Metanorms, Topologies, and Adaptive Punishment in Norm Emergence

Mahmoud, Samhar

Awarding institution:
King's College London

Metanorms, Topologies, and Adaptive Punishment in Norm Emergence

by

Samhar Mahmoud

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the

Department of Informatics

in the

School of Natural & Mathematical Sciences

April 2013

Abstract

Norms provide a means to regulate the behaviour of the members of a society, organisation or system. While much work has been done on various aspects of norms, normative systems and normative behaviour, this work has been limited in several respects. In particular, the problems of norm emergence have only recently begun to be considered, with existing work adopting only simple structural models. This relates to two crucial issues that have not adequately been addressed. First, existing models of norms assume that sanctions are static, and do not change in relation to relevant information about the violator, the situation or the history. Yet, typically there is information available that can significantly impact on the nature of such sanctions, and can even allow sophisticated sanctioning structures that achieve more effective regulation. Second, work on norm emergence has typically assumed simple topological structures of agents, if any at all, yet real computational systems, in which norms are relevant, such as peer-to-peer systems and wireless sensor networks, may have topologies of varying degrees of sophistication. These topologies constrain potential relationships between agents, limiting the observation of violations, and possibly also limiting the kind of sanctions than may be imposed. In this thesis, therefore, we seek to address these problems in support of more effective norm-regulated systems, by developing mechanisms that can incentivise cooperative behaviour in societies of self-interested agents.

Acknowledgements

The first and very special thanks is for my outstanding supervisor Professor Michael Luck *who has set an example for me that I hope to match some day. Thanks for his expertise, understanding, and patience in all stages of my PhD. I appreciate his vast knowledge and skills in many areas, and his continuous help and support which helped me shape my ideas and focus my interests. In addition to being my PhD supervisor, Mike has been the first person I would seek guidance from regarding personal life matters, which he has always welcomed and been of great help.*

My gratitude goes out as well to both Dr. Jeroen Keppens *and Dr.* Nathan Griffiths*, whose precise comments and experience helped putting me on the right track and produce the work in this thesis. I must also acknowledge my friend Dr.* Daniel Villatoro *with whom I had the great opportunity to collaborate, which helped taking my work forward. Similarly, I want to thank Dr.* Gareth Tyson *for sharing his experience in the the domain of P2P file sharing, which helped in forming the case study introduced in this thesis. Appreciation also to my colleague and friend* Matthew Shaw *who has always been there to discuss and share thoughts.*

My gratitude is extended to all the lecturers, colleagues and administration staff in the department of Informatics at King's College London for the amazing, friendly and comfortable environment that they have created for everyone, which helped producing my work in the best possible way. In particular, thanks to Dr. Simon Miles*, Dr.* Sanjay Modgil*, Dr.* Adel Taweel*, Dr.* Elizabeth Black*, and Professor* Peter McBurney*. Thanks to my colleagues:* Andrada, Christos, Gbolahan, Haixiao, Ingrid, Laura, Maria, Martin, Padmaja, Shekoufah, Valeriia, Vida*, and many others who I have worked with. Thanks for* Claudia Mazzoncini *whose valuable work has been of help to everyone in the department.*

Thanks to all my friends who gave me only good moments and made my life, far away from home, easy. To my friends: Ammar, Feras, Franchesco, Jaz, Modar, Muhaned, Osama, Rami, Radhicka, Rawad, Reem, Saeed, Shady, Suzanna. Thanks to my family for their unbounded love and continuous support. To my family: My father Ali, my mother Naelah, My sisters Hyam and Lama, my brother Ghiath, my precious nephews Mufid, Majed and Karam, my inlaws Wafik, Olga, Aous, Shaam, Linda, Lida, Stewart, Sasha and Anna, and my cousins Nawar and Nebras

Saving her to the last, I wish to acknowledge my wife, colleague and best friend, Lina Barakat*, without whose love, encouragement and proofreading assistance, I would not be where I am now.*

To my beloved ***Syria*** ...

Contents

List of Figures

Chapter 1

Introduction

Agent-based computing is one of the most active research areas in computer science, and has been gaining much attention in recent years. Most, if not all, practical agent systems, are multi-agent systems, with agents interacting, in line with the requirement for social ability. In many current systems, such interactions take place within closed environments in which the participating agents are known in advance, and in many cases are benevolent in the sense that they are guaranteed to assist others. However, this is not a realistic assumption for real-world problems, and for the increasing trend towards open and dynamic computational systems in which agents can enter and leave at any time. This introduces problems in that when agents are not benevolent, they can take actions that are not constructive, or indeed are malicious, and cause difficulties for the system or society as a whole.

In response, such systems or societies often use constraints, rules or policies to be followed by their participants, with the aim of avoiding such malicious behaviour. These constraints, rules or policies, which restrict the behaviours of agents, are known as *norms*, and they have gained much recent attention in the literature. Yet while there

has been a significant amount of work in this area, there are still several aspects that have not merited as much coverage. In particular, while representations for norms have been studied extensively, and while there has been some work on the decision-making process of agents governed by norms, there has been only limited consideration of the issues surrounding the *emergence* of norms in a population of agents, where a pattern of behaviour (or norm) arises from the direct interactions of those agents rather than being imposed on them.

The rest of the introduction is organised as follows. Section 1.1 introduces the multi-agent systems paradigm of interest in this thesis. In Section 1.2, we describe some relevant scenarios illustrating the problems involved in regulating distributed systems, which are followed, in Section 1.3, with specific problem characteristics that are concluded from these scenarios. The research aims of this thesis and the methodology followed to achieve these aims are introduced in Sections 1.4 and 1.5, respectively. Section 1.7 provides the list of publications generated from the work conducted in this thesis. Finally, the thesis structure is presented in Section 1.8.

1.1 Agents and Multi-Agent Systems

Despite the large amount of research that has been undertaken in the field of agents and multi-agent systems, there is no specific *definition* of an agent that is commonly agreed. However, it is generally accepted that agents can be understood as computer systems that are situated in some environment, and capable of autonomous action in this environment in order to meet their design objectives [148]. Moreover, the key properties characterising agents, are enumerated by Wooldridge and Jennings [149] as being: *autonomy*, by which agents operate in their environment without any human interference, and have control over their behaviour; *reactivity*, by which agents perceive

their environment and respond in a timely fashion to changes that occur within it; *social ability*, by which agents interact with humans and other agents and even cooperate in some cases; and *proactiveness*, by which agents do not just act in response to their environment, but instead have their own goals and take any opportunity to achieve these goals.

Since agents are social entities, and operate in environments containing other agents with which they will generally be expected to interact, and since most substantial problems are usually beyond the capabilities of one individual agent, *multi-agent systems* in which agents work together, offer a solution. Many definitions have been proposed, but we introduce the most common one. Multi-agent systems are collections of intelligent agents interacting with each other in order to achieve their own goals or to solve a common problem in their environment [148]. Of course, since agents are autonomous, and have their own goals, they may not always agree, and may choose not to collaborate. Moreover, in some cases, their interests may conflict, so that what benefits one may constrain another.

In closed systems, in which agents can be carefully controlled, or in which they are guaranteed to be benevolent and always help others, this is not a problem. However, in open and dynamic systems, where agents can join and leave, and there are no constraints on membership, the lack of any control may give rise to ineffective or inefficient overall system behaviour. Thus, there is a need for some means to regulate individual agent behaviour in support of overall system performance. Norms provide just such a means that can help to avoid conflicts between self-interested agents in this context.

Norms can be introduced into such systems in various ways. They can be imposed by a central *authority* that has the power to impose the rules or constraints that are

necessary to regulate the system. As opposed to this *top-down* approach, in a *bottom-up* approach, a norm emerges through the interaction of the agent population of these systems and thus is driven by the agents themselves without any central control.

1.2 Regulating Distributed Systems

In many application domains, engineers of distributed systems may choose, or be required, to adopt an architecture in which there is no central authority, and the overall system consists solely of self-interested autonomous agents. The rationale for doing so can range from efficiency reasons to privacy requirements. In order for such systems to achieve their objectives, it may nevertheless be necessary for the behaviour of the constituent agents to adhere to *norms*, as introduced above. In peer to peer file sharing networks, for example, we require (at least a proportion of) peers to provide files in response to the requests of others, while in wireless sensor networks nodes must share information with others for the system to determine global properties of the environment. However, there is typically a temptation in such settings for individuals to deviate from the desired behaviour, which is known as the problem of *free riding* behaviour. For example, to save bandwidth peers may choose not to provide files, and to conserve energy the nodes in a sensor network may choose not to share information. It is therefore important to minimise the temptation for agents to deviate from the desired behaviour, and to encourage the emergence of cooperative norms.

We consider the problem of free-riding in these scenarios in more detail below.

1.2.1 Peer to Peer File Sharing

Schollmeier [114] considers a peer to peer system to be a distributed system that consists of members, each of which share some resources (hardware, software or information)

with other members. While there is much work that addresses the issue of free riding behaviour in P2P file sharing systems [1, 48, 52, 67, 105, 153], here we focus on it in the context of the specific case of the *Gnutella* system.

Gnutella is a P2P file sharing network, in which each peer plays the role of both client and server. As a client, a peer requests files from others, while as a server a peer provides files to others. When a peer needs access to a file in the network, it forms a query regarding the desired file and passes this query to its neighbours. If the neighbour has the file, it replies to the request. If not, the neighbour passes the request to its own neighbours and passes the response of those neighbours back to the requesting peer. When the original peer receives a file, it should make it available to others. A peer does not pay anything to access files and there is no limit on the amount of files the peer can access, nor on the proportion of the files it shares. Therefore, it might be rational for peers to not waste their bandwidth responding to other requests as they can access different files on the network without sharing any of their own files. This is known as the problem of *free riding*. As shown by Adar and Huberman [1], 70% of Gnutella peers share no file (they receive files from the network without sharing in return).

Importantly, it is the absence of any central authority that can monitor the behaviour of all peers, and react to those that do not share the files they download, that makes it easy for free riding behaviour to be established. However, such a central authority is not sensible, because of the vast number of participants that usually participate in such systems, making the complexity of monitoring interactions between participants far too complex.

In some recent efforts to try overcoming the issue of free riding [6, 11, 59, 144], peer punishment has been used, in which peers themselves apply some penalty to those that do not share the files they download, such as imposing a fine, limiting interactions or even ceasing interaction entirely. However, since peers are only connected to a small

subset of those in the system, rather than to all other peers, there are constraints on communication or interaction paths, which can make it difficult for the peers themselves to effectively monitor others, apply this approach and bring about better overall performance.

1.2.2 Wireless Sensor Networks

A wireless sensor network (WSN) is a network of connected, physically small, and low cost sensors that sense their surrounding environment and exchange the sensed data in a wireless manner. These abilities allow this kind of network to be used in various civilian or military applications such as fire detection [104], health care [89], weather forecasting [84] and tracking systems [55, 66].

Suppose we have a WSN for tracking objects, in which the objective is to track a moving object using a number of sensors that have limited sensing range. Here, sensors are distributed to cover a large area according to a certain topology. Each sensor monitors the sub-area for which it is responsible. If an object appears in the sensor area, the sensor passes information on that object to its neighbouring sensors, which, in turn, pass the information to their neighbours until the information reaches a predefined location. However, because sensors have limited power, or because they might belong to different companies, they can choose not to cooperate with other sensors and refuse to pass on information, or even not to track objects within their range. In this case, the network as a whole cannot be trusted to track objects. Exactly the same problems apply here as in the previous peer to peer scenario.

1.3 Problem Characteristics

Drawing on the above scenarios, the problem we are concerned with can be generalised as follows. We have a multi-agent system, in which the constituent agents (e.g. peers or sensors) are autonomous, with no central authority controlling the whole system, and yet the agents must exchange information between them in support of some overall system goal. Here, there is an implicit norm of exchanging this information. However, since for any individual agent there may be a decrease in utility from exchanging information, agents may choose not to comply with this norm, and hence some means of incentivising appropriate behaviour is needed, possibly through sanctions. Importantly, the system topology, which defines the relations between agents, also limits their interactions and, in addition, constrains what can be observed, making the detection of violation, as well as punishment of it, much more difficult. The problem can thus be broken down into the following characteristics.

- There is a group of self-interested autonomous agents,
- in an open environment in which agents can join and leave at any time.
- There is no central authority, so that
- agents may or may not comply with the desired behaviour.
- Finally, agents are connected via a topology that
- constraints interactions among agents and
- observation of their violation or compliance, and
- their ability to apply sanctions to each other.

1.4 Research Aims

To summarise, norms provide a means to regulate the behaviour of the members of a society, organisation or system. While much work has been done on various aspects of norms, normative systems and normative behaviour, it has been limited in several respects. In particular, the problems of norm emergence have only recently begun to be considered, with existing work adopting only simple structural models. This relates to two crucial issues that have not yet adequately been addressed. First, existing models of norms assume that sanctions are static, and do not change in relation to relevant information about the violator, the situation or the history. Yet, typically there is information available that can significantly impact on the nature of such sanctions, and can even allow sophisticated sanctioning structures that achieve more effective regulation. Second, work on norm emergence has typically assumed simple topological structures of agents, if any at all. Yet real computational systems in which norms are relevant, such as the peer-to-peer systems and wireless sensor networks discussed above, may have topologies of varying degrees of sophistication. These topologies constrain potential relationships between agents, limiting the observation of violations, and possibly also limiting the kind of sanctions that may be imposed. In this thesis, therefore, we seek to address these problems in support of more effective norm-regulated systems, by developing mechanisms that can incentivise cooperative behaviour in societies of self-interested agents.

More specifically, in this context, the aims of this thesis are as follows:

- to provide a means of supporting norm emergence in complex distributed systems without the need for a central authority;
- to ensure that such norm emergence is effective over various types of interaction topologies that are known to describe the structure of real world systems; and

- to consider the efficiency of norm emergence, by adopting techniques that apply only the effort necessary to bring it about.

1.5 Methodology

In order to achieve these aims surrounding norm emergence in complex non deterministic domains, this thesis adopts an empirical approach for the purpose of validating the theoretical concepts and mechanisms introduced throughout the thesis.

In each contribution chapter, a particular mechanism incorporating a solution to a specific problem is introduced. This is followed by a set of experiments assessing the effectiveness of the mechanism proposed. For each experiment, three different steps are distinguished: the experimental set-up, which introduces the different parameters needed and used; the *main* experimental results, which provide an indication of the effectiveness of the mechanism under investigation; and lastly some *supportive* experimental results, which are needed in order to provide a better explanation of the main results obtained.

1.6 Simulation Platform

All experimental results presented through this thesis were obtained using a simulation built with the *Java* programming language. The simulation assumes asynchronous execution where agents take turns to make decisions and to observe decisions made by other agents. The duration of execution of these experiments varied between hours and days depending on the sophistication of the model being evaluated. All experiments were run on a single PC with 8GB of *RAM* and a Quad Core 2.66 GHz processor. The main constraint of our experiments was CPU speed, rather than memory. We were

therefore able to run multiple experiments simultaneously, utilising all four cores of our CPU.

1.7 Publications

The work in this thesis has appeared in several publications in various venues, as detailed below.

The content of Chapter 3 has been published in:

- S. Mahmoud, N. Griffiths, J. Keppens, and M. Luck. An analysis of norm emergence in Axelrod's model. In *NorMAS'10: Proceedings of the Fifth International Workshop on Normative Multi-Agent Systems*. AISB, 2010
- S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. An analysis of norm emergence in axelrod's model. In *EUMAS'10: Proceedings of the 8th European Workshop on Multi-Agent Systems*, 2010

The content of Chapter 4 has been published in:

- S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Overcoming omniscience in Axelrod's model. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, volume 3, pages 29–32, 2011
- S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Overcoming omniscience for norm emergence in Axelrod's metanorm model. In B. van Riemsdijk S. Cranefield, J. Vazquez-Salceda and P. Noriega, editors, *Coordination, Organizations, Institutions and Norms in Agent Systems IV*, volume 7254 of *Lecture Notes in Computer Science*, 2012

The content of Chapter 5 has been published in:

- S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Norm establishment via metanorms in network topologies. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, volume 3, pages 25–28, 2011

- S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Establishing norms for network topologies. In B. van Riemsdijk S. Cranefield, J. Vazquez-Salceda and P. Noriega, editors, *Coordination, Organizations, Institutions and Norms in Agent Systems IV*, volume 7254 of *Lecture Notes in Computer Science*, 2012

- S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Overcoming hub effects in scale free networks. In *Proceedings of the Fourteenth International Workshop on Coordination, Organisations, Institutions and Norms (COIN 2012)*, pages 136–150, 2012

- S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Norm emergence through dynamic policy adaptation in scale free networks. In *Coordination, Organizations, Institutions and Norms in Agent Systems V*, 2013. to appear

The content of Chapter 6 has been published in:

- S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Efficient norm emergence through experiential dynamic punishment. In *Proceedings of the 20th European Conference on Artificial Intelligence*, pages 576–581. IOS Press, 2012

- S. Mahmoud, D. Villatoro, J. Keppens, and M. Luck. Optimised reputation-based adaptive punishment for limited observability. In *Sixth IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, 2012

1.8 Thesis Structure

The rest of the thesis is organised as follows. Chapter 2 reviews related background to the area of norms in general and norm emergence in particular. Chapter 3 provides a deep analysis of Axelrod's model, identifying specific problems that prevent using it for regulating distributed computational systems. Chapter 4 introduces a model that omits some of the unrealistic assumptions of Axelrod's model, including the need for central authority, and knowledge of the private strategies of others. Chapter 5 incorporates interaction topologies into the model, and handles the problems arising from such incorporation. Chapter 6 is concerned with the adaptive punishment technique, allowing various levels of punishment to be applied according to the case at hand. Chapter 7 performs a case study analysis in the domain of peer to peer file sharing, to evaluate the developed models in a specific real-world context. Finally, Chapter 8 concludes the thesis with a summary and possible future work.

Chapter 2

Background

2.1 Introduction

In open societies, there are a number of self-interested agents whose behaviours might deviate from those that are expected. There is therefore a need to regulate such behaviours in order to seek to provide some assurance that they deliver what is expected from them. Norms can be used as a mechanism to avoid such deviations and to avoid conflict within these societies.

In this chapter, we explain the concept of norms and discuss different aspects related to them. We structure our discussion, and these aspects, along the lines illustrated in Figure 2.1, with each aspect being covered in a different section. First, in Section 2.2, we introduce some definitions of norms for the purpose of showing the various perspectives, after which we provide an analysis of norm categories in Section 2.3. Section 2.4 reviews work concerned with the components of norms and Section 2.5 reviews some logics used to specify norms. Section 2.6 explains how norms influence the behaviours of agents, while Section 2.7 then outlines some work that handles the

issue of norm enforcement. Finally, Section 2.8 discusses the issue of norm emergence, which is the focus of this thesis, before concluding in Section 2.9 with a discussion of some outstanding problems.

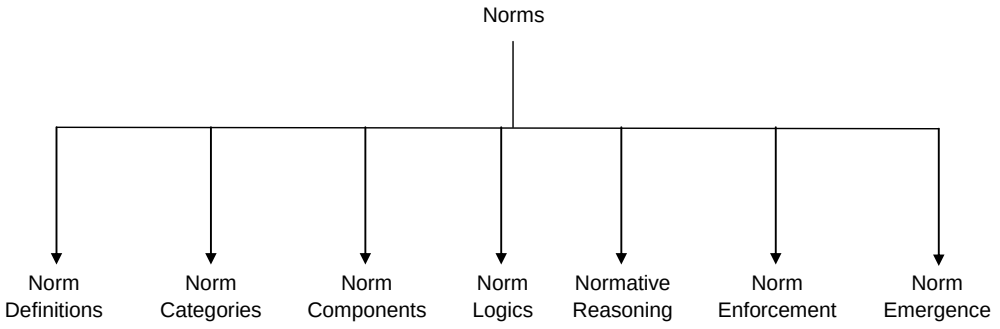


FIGURE 2.1: Review outline

2.2 Norm Definitions

To start, and to allow a better understanding of the meaning of the term norm, we provide in this section some definitions of norms proposed by different researchers from different backgrounds. For example, the sociologist, Gibbs, defines a norm as "a collective evaluation of behaviour in terms of what it ought to be; a collective expectation as to what behaviour will be; and/or particular reactions to behaviour, including attempts to apply sanctions or otherwise induce a particular kind of conduct" [44]. Gibbs assigns norms to prescribing the required behaviour of society members as well as the expected behaviour of these members. A norm in his view also defines a method of reaction to a specific behaviour, which might include rewards or punishments as a result, and can be used as means of persuasion to adopt a specific behaviour.

From a different perspective, Therborn [126] presents an overview of the *role* of norms. He particularly emphasises the meaning of normative actions or actions that are related

to, or caused by, norms. A normative action according to Therborn can be of two kinds: *teleological* or *emotional* actions. A teleological action is an one based upon the desire to do the right thing rather than the thing that leads to ends or goals, while an emotional action is one that is based on the thing that leads to or is caused by an emotion.

More recently, López y López and Luck [150] proposed a more comprehensive definition of norms within their normative framework. According to them, norms facilitate mechanisms to drive the behaviour of agents, especially in those cases where their behaviour affects other agents. Norms can be characterised by their prescriptiveness, sociality, and social pressure. In other words, a norm tells an agent how to behave (prescriptiveness) in situations in which more than one agent is involved (sociality), and since it is always expected that norms conflict with the personal interest of some agents, socially acceptable mechanisms to force agents to comply with norms are needed (social pressure).

2.3 Norm Categories

Here, we introduce the different kinds and categories of norm. Many researchers (e.g., [14,126,130]) have tried to analyse and categorise norms. Some analyse these categories from the philosophical prespective such as Tuomela [130, 131] who categorised norms based on the reason of their creation (created by a certain authority or based on mutual beliefs) and the procedure of this creation (formal or informal). Others, such as Therborn [126], and Boella and van der Torre [14] discuss norms and categorise them based on the roles they play. Nevertheless, all agree more or less on four general categories of norms, which are *constitutive* (institutional facts), *regulative* (governed behaviours), *procedural* (rules creation constraints) and *distributive* (sanctions) norms. These categories are shown in Figure 2.2 and are explained in what follows.

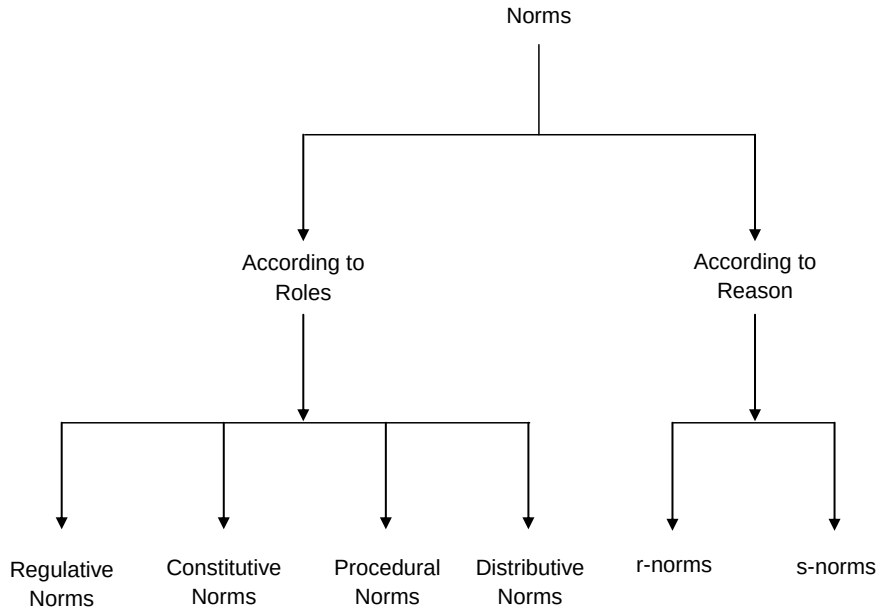


FIGURE 2.2: Norm Categories

From a philosophical point of view, Tuomela [130, 131] classifies social norms as one of two kinds of rules (*r-norms*) and proper social norms (*s-norms*). In this view, r-norms represent implicit or explicit agreements between different agents of a society and should be created by a certain authority of this society, while s-norms are based on a mutual belief. Tuomela divides r-norms and s-norms into two further divisions: r-norms can be formal or informal, where formal rules are in explicit written forms and include sanctions, while informal rules are usually communicated orally and can include informal sanctions. On the other hand, s-norms are divided into *conventions* that concern the whole society, and *group-specific* norms that concern a group of agents in the society.

From another perspective, Therborn [126] distinguishes between three kinds of norms: *constitutive* norms define a system of action and an agent's membership in it; *regulative* norms describe the expected contributions to the social system; and *distributive* norms

specify how rewards, costs, and risks are allocated within a social system.

Within the context of normative multiagent systems, Boella and van der Torre [14] agree with Therborn on two of the three categories, but replace the distributive norms with a new category, *procedural* norms. They state that a norm can fit into one of three categories, as follows.

- Regulative norms specify the rules that govern the behaviours of any member of the system in terms of obligations, prohibitions and permissions.
- Constitutive norms are used to identify institutional facts that exist in a system. These kinds of norms are based on the notion of *counts-as*, which can be used to state that a concept can be considered the same as another one, such as synonyms in English. The use of the counts-as notion allows construction of a relation between different concepts which, according to Boella and van der Torre, can lead to the establishment of an ontology that supports regulative norms.
- Procedural norms can be seen as a nesting of regulative and constitutive norms. The role of these norms is to guarantee the satisfaction of social order from other kinds of norms. This can be accomplished by providing agents with mechanisms for enforcing and detecting violation.

In addition, procedural norms can also be distinguished as a major component of political systems. Lawrence [71] states that procedural norms are rules governing the way in which political decisions are made; they are not concerned with the content of any decision except one which alters decision-making procedures. From a completely different point of view, Dignum and Kinny [35] specify the difference between norms in general and obligation in particular. They specify that the effects of satisfying an obligation are usually directed toward another individual of the society and in most cases an obligation is assigned with punishments to assure its enforcement. Conversely,

general norms are usually applied to an agent by the society the agent belongs to, and its satisfaction benefits the whole society much more than a single member.

2.4 Norm Components and Representation

Having defined norms and identified their categories, in this section we consider the potential components of a norm and review some work that addresses the issue of norm representation. If the purpose of norms within a society is to regulate the behaviours of the society's members, then they need to be understandable by these members, which in our case are agents. Therefore, norms need a structure that specifies their components and the relations between these components.

López y López and Luck [150] propose that each norm should consist of:

- normative goals representing the goals prescribed by a norm;
- addressee agents, which are the agents that must comply with the normative goals;
- beneficiary agents that might benefit from norm compliance;
- context, the circumstances under which the norm is activated;
- exception cases in which agents are not required to fulfil the norm;
- punishments, which are penalties applied to agents that do not satisfy the normative goals; and
- rewards, which are given to agents that comply with norms.

Vázquez-Salceda et al. [138] focus on how norms should be operationally implemented in multiagent systems from an institutional perspective. Their work introduces a classification of the different attributes of norms, which can be characterised by whether:

- they refer to a state or an action;
- they are conditional;
- they include a deadline; or
- they are norms concerning other norms.

In order to express such norms, they propose a language that makes use of deontic concepts. Building on this, Grossi [50] also takes into account the ontological aspects of norm implementation. He introduces a notion of contextual ontology and suggests that institutions use it to relate the abstract concepts in which their norms are formulated to their concrete application domain. In his view, many institutions can implement the same set of norms in different ways as far as they presuppose divergent ontologies of the concepts in which that set of norms is formulated.

Okuyama et al.'s approach [98] shares some elements with the representation suggested by López y López and Luck. Okuyama et al. propose the notion of *situated* norms, which refer to norms that can be applied in a certain place only. In their representation, a norm consists at least of the following elements:

- *type*, which refers to the type of the norm and its level of importance, such as obligation, warning or direction;
- *issued by*, which indicates the authority that issued the norm;

- *norm string*, which contains the main information of the norm, including the agent to which the norm applies, in addition to the action that is required by the norm.
- *placement*, which refers to the place in which the norm is applied;
- *condition*, which specifies the condition under which the norm is triggered; and
- *id*, which uniquely identifies the norm in the whole system.

2.5 Norm Logics

In terms of representation, logic has been used extensively to formally represent norms and specify their components. Given this, and although logics are not specifically relevant to the focus of this thesis, we provide a brief overview of relevant work on logics of norms for completeness. In particular, there are logics specifically proposed to reason with and about norms, *deontic logics* [87, 135], while some researchers have also tried to formalise norms by using of the expressive power of other logics, such as *defeasible logic* [4, 5, 10, 46, 47] and *input/output logic* [12, 14, 85]. Others still have tried to express norms via integrating two logics together, such as *defeasible deontic logic* [97, 107, 136]. In what follows, we introduce some aspects of both deontic and defeasible logic in relation to norms for the purpose of completeness (as indicated above) and context.

2.5.1 Deontic Logic

Deontic logic is the branch of symbolic logic that is concerned with what is obligatory, permitted and prohibited. Deontic logic has been an area of discussion and argumentation since the fourteenth century, but the first significant phase in the development

of deontic logic is the work by Mally [72], since he was the first to propose a formal system of deontic logic based on a system of propositional calculus. Based on Mally's work and the early development of the alethic modal logic, von Wright published his system of deontic logic in 1964 which, along with Mally's work, has led to what is known as Standard Deontic Logic (SDL). SDL is the most cited and studied system of deontic logic [87] and is based on propositional logic. Deontic logic has been an active area of research and has had many extensions. Some researchers [20, 127, 135] have extended it to include conditions that can be defined as $OB(A/B)$, which means that A is obliged only if B is true. Others introduced the idea of *contrary to duty* obligations [21, 102, 128, 137], which tells us what comes into force when some other obligations are violated.

2.5.2 Defeasible Logic

Defeasible logic was originally proposed by Nute [96] with particular concern for efficiency and implementation. Nute's logic has gained more attention over the years and has been extended and developed by many researchers (e.g. [4, 5, 10]).

Apart from the representation of rules, defeasible logic allows the assignment of priorities to rules in order to solve conflicts between them. A conflict can occur between two rules when one of them includes A in its conclusion, while the other includes $\backsim A$. In this case, defeasible logic chooses the rule with the higher priority. Governatori and Rotolo [46, 47] extend defeasible logic in order to allow the combination of mental attitudes (beliefs and intentions) and obligations. They use a labelling mechanism on defeasible rules to allow the representation of obligations such as a rule that specifies that if a customer has purchased a good x of price y, then he is obliged to pay y.

2.6 Normative Reasoning

Having introduced topics related to defining and representing norms and their internal components, the rest of this chapter explains how norms are used in the context of agents and multi-agent systems and we start in this section by describing the process of normative reasoning, and the relation between agent reasoning and norms.

In order for agents to act, they must reason about achieving their goals using the plans that are available to them, but agent reasoning is also affected by the existence of norms. Normative reasoning refers to the reasoning that takes into consideration norms in the society in which such agents act.

Across the literature on norms, three main processes can be identified within an agent's normative reasoning that have been identified in the literature: *norm recognition*, *norm adoption* and *decision making*. When an agent joins a society, it needs to communicate the norms of this society, which requires distinguishing these norms from other social elements, such as messages or queries. This is what the process of norm recognition is concerned with. After recognising norms, the agent needs to identify the subset of norms that are actually directed towards it, which is the task of the norm adoption process. Now, since adopted norms might conflict with each other and might even conflict with some of the agent's goals, the agent needs to decide which norms it is going to comply with within the decision making process. The following three sections address these normative reasoning processes in more detail and introduce some of the work from the literature that addresses them.

2.6.1 Norm Recognition

Norm recognition refers to the process that allows an agent to decide if what it has been told is a norm is actually a norm from its own perspective. Within their architecture

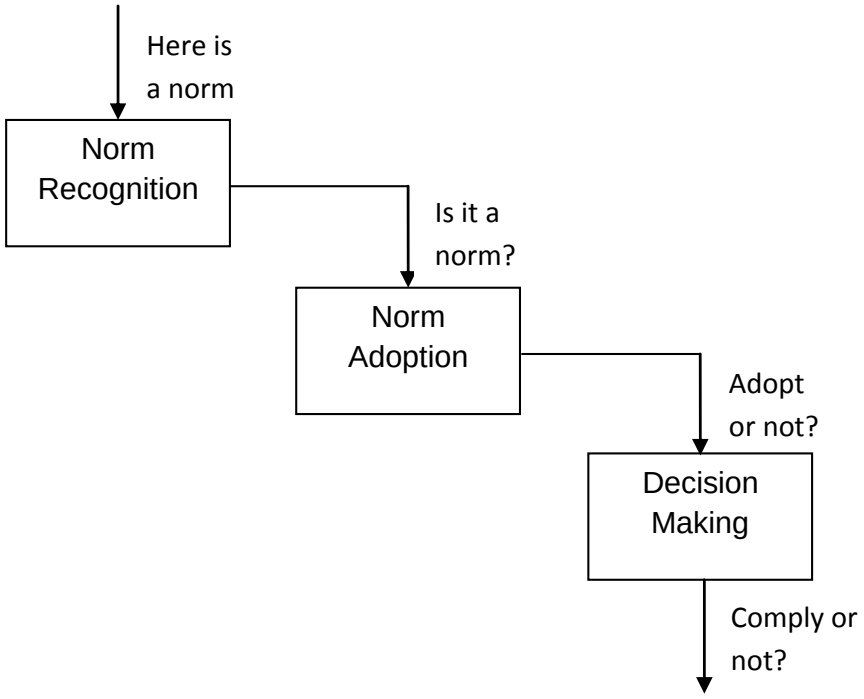


FIGURE 2.3: Normative Reasoning Main Processes

(see Section 2.6.4 for more details), Andrighetto et al. [2] claim that the norm *recogniser* plays a very important role within the agent reasoning cycle, because it allows an agent to distinguish between norms and other forms of social inputs such as ordinary requests and messages.

According to Conte et al. [25], a new norm arises because of different reasons, summarised as follows.

- A new norm can be recognised if it is actually a version of an existing norm. An example of this is when the norm is an instantiation of an existing norm or an interpretation of it.
- The agent can accept a new norm if the issuer of what is said to be a norm is known to be a normative authority that is allowed to issue norms.

- A new norm can be evaluated against the motivation of the issuer. If the norm is issued because of the issuer's self interest but it has no utility for the society, then it can be discarded, and adopted otherwise.

2.6.2 Norm Adoption

When an agent recognises a new norm, it decides to accept this norm if it believes that the new norm concerns or is directed towards its behaviour. This is known as norm adoption. Conte et al. [25] state that an agent accepts (adopts) a norm only if it believes that this norm helps either in a direct or indirect way to achieve one of its goals. Based on this, the agent forms a normative goal that results from its decision to adopt the norm, but it does not make the decision to *comply* with this norm. In addition, López y López et al. [73] specify that in order for an agent to adopt a norm, the agent should be recognised as an addressee of the norm.

Because agents are expected to join different societies and interact with other members of these societies, agents are also expected to encounter different norms that are applied in each of these societies. However, adopting these norms may cause conflicts with an agent's current norms, which raises the issue of *norm consistency*, where a set of norms is consistent if there are no conflicting norms within this set. Within the NOA architecture [65], Kollingbaum and Norman discuss the issues of norm adoption and consistency. They specify that adoption of an obligation can cause conflicts with other norms, which can be the case if a plan to fulfil the new norm is either forbidden or contradicts a currently active obligation. Conversely, the adoption of a new prohibition can cause inconsistency if it forbids the demands of an active obligation.

In order to help an agent decide whether to adopt a new obligation that conflicts with the current set of norms, Kollingbaum and Norman introduce the concept of norm

consistency, which can take one of three forms: *strong consistency*, *weak consistency* and *inconsistency*. Norm consistency has a major impact on the process of selecting a plan to fulfil a newly adopted norm, as it indicates whether a norm causes a conflict with an agent's current norms. Strong consistency indicates that adopting a new obligation causes no conflicts with the agent's current set of norms and the agent is free to choose any plan that is suitable for achieving the purpose of the new obligation. Weak consistency indicates that among the set of plans that are suitable for fulfilling the new obligation, there exist some plans whose execution will cause conflicts with the agent's current set of norms, such as a plan that involves the execution of an action that is forbidden by another norm. Here, the agent still has some plans that can be executed without any conflicts, so being more careful when selecting the plan will avoid the conflict. Inconsistency indicates the non-existence of any plan that can be executed to fulfil the new obligation without causing conflicts. In this case, the agent needs to use conflict resolution strategies in order to decide which of the conflicting norms the agent will comply with.

2.6.3 Decision Making

Decision making is a critical phase of normative reasoning, as an agent decides within this phase if it is going to comply with a norm. Whatever the decision, this can have a major impact on the agent's behaviour. If an agent complies with the norm, then some of its goals might conflict with the norm and as result the agent will not be able to achieve any of these conflicting goals. Conversely, if the agent refuses to comply with the norm then some punishments may be applied to the agent, which in turn can affect the achievement of some of its goals.

There have been a few attempts to deal with the decision making phase of normative reasoning [13]. Most are concerned with decisions based on the existence of conflicts

between different norms, or between norms and goals, considered as a part of the norm adoption phase, as explained in the previous section.

Conte et al. [25] state that an agent decides to comply with a norm based on different criteria. It might refuse to comply with a norm if the norm conflicts with more important goals or with other norms that the agent has already made a decision to comply with. Alternatively, the agent might decide to fulfil a norm because of the *guilt* resulting from not fulfilling the norm or because of the consequence of violating norms. Kollingbaum and Norman [65] propose different strategies to resolve conflicts between different kinds of norm and decide which norm to comply with. For example, an agent can decide to comply with a norm that is issued by a source whose *social power* is higher, or to adopt a norm that has been activated more recently.

2.6.4 EMIL-A

In EMIL-A, Andrighetto et al. [2] explain the main phases that norms go through in order to evolve from the environment into the agent's internal state. In their view, which agrees with the categorisation shown above, these phases involve recognising norms, adopting them and deciding whether to comply with them. As shown in Figure 2.4, EMIL-A consists of several distinct components.

- There are four procedures:
  - *norm recognition*, which is responsible for recognising a new norm;
  - *norm adoption*, which determines whether to adopt the norm or not;
  - *decision making*, which determines if the agent will intend to comply with the norm; and
  - *normative action planning*, which produces a plan that conforms with the purpose of the norm.

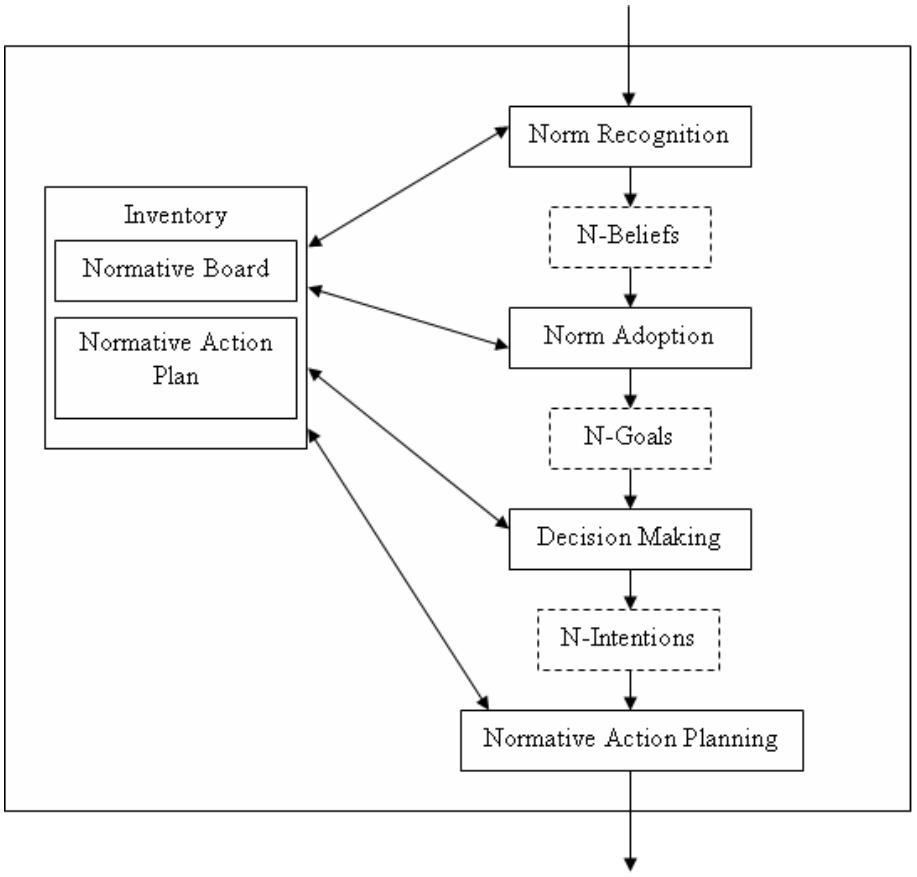


FIGURE 2.4: EMIL-A Architecture (From [2])

- There are three mental objects:
  - *normative beliefs*, which result from norm recognition;
  - *normative goals*, which result from norm adoption; and
  - *normative intentions*, which result from decision making.
- Finally, there is an inventory that contains a *normative board*, which is a set of existing norms and normative information that the agent already has, and a repertoire of normative action plans (plans that consist of actions that respect norms).

The resulting behaviour of EMIL-A can be of two types: the agent can either comply with the norm or violate it. The latter may trigger some defence mechanisms that seek to enforce the norms. Apart from the architecture design, Andrighetto et al.'s work [2] also focuses on how EMIL-A allows a new norm to be perceived and established as an instance of an existing norm, which is a responsibility of the norm recognition component. EMIL-A provides a valuable contribution in specifying the main processes of normative reasoning (reasoning with respect to norms) but it does not really show *how* most of these process actually work in detail.

2.7 Norm Enforcement

As mentioned in the introduction of this thesis, one of the most salient attributes of agents is autonomy, which means that their behaviour can be unpredictable and, as a result, complying with norms is not guaranteed. However, norms exist as rules for the benefit of societies, and violating them might negatively affect others in these societies. For this reason, there is a need, on the part of the society, to reduce or even to prevent the number of norm violating cases by what is known as *norm enforcement* [22, 39, 58, 129]. The use of sanctions against a norm violator can be seen as a *norm enforcement mechanism*, because an agent's decision to comply with a norm should take into consideration any penalties incurred by violating it.

Posner and Rasmusen [101] discuss different types of sanctions and categorise them as follows.

- Automatic Sanction: The actions of the violator include in themselves some harm that affects the violator. An example of this is that when a person drives his car on the wrong side of the road, he might crash into another car and harm himself.

- Guilt: The violator feels guilty about his actions, because of his level of education and ethics.
- Shame: The violator feels that his actions affected his reputation negatively in the eyes of other members of his society.
- Informational Sanction: Informational sanctions are applied by informing the members of the society about undesired actions of the violator.
- Bilateral Costly Sanction: Bilateral costly sanctions involve a punishment that is applied by another member of the society. This type of sanction is costly because it requires another member of the society to apply the sanction to the violator. It also involves effort to monitor and detect the violation.
- Multilateral Costly Sanction: Multilateral costly sanctions involve a punishment that is applied by more than one member of the society.

The issue of using sanctions to enforce norms was addressed by others, such as López y López et al. [151], who define the notion of *interlocking* norms. Two norms can be interlocking, if satisfying or violating one (the primary norm) triggers the activation of the other (the secondary norm). López y López et al. suggest applying the notion of interlocking norms to norm enforcement by specifying that reward or punishment norms are represented as secondary norms for the primary norms that the rewards or punishments are assigned to.

An example application that uses sanctions as a reaction to norm violation is described by de Pinninck et al. [29], who use Gnutella, a pure peer-to-peer file sharing application, which allows different peers to search for files hosted by other peers and to download these files to their own hard drive. However, Gnutella also allows a peer to join the network and keep downloading files without benefiting other members of the network, for example by sharing some files on the peer's hard drive. To prevent this problem

of allowing peers to consume the network's services without participating by providing any service, de Pinninck et al. suggest adding a norm that specifies that any agent that needs to download a file from another agent should share some files on its own hard drive. To ensure compliance, they propose ostracising the violator, so that no other member of the society interacts with it and, as a result, it has no access to any of the network's files. Ostracising the violator is accomplished through the spread of its negative reputation over the network.

2.8 Norm Emergence

As mentioned earlier, the existence of norms does not mean that agents will always comply with them. Within the literature, norm compliance can be established using two different approaches, top-down and bottom-up. In the top-down approach, a norm is imposed by some kind of authority, which is responsible for monitoring compliance with the norm. In the bottom-up approach, agents become aware of the existence of a norm through their interactions with each other, and it is the responsibility of every agent to monitor compliance with this norm. Another way to refer to the top-down approach is as *normative reasoning*, which is addressed in Section 2.6, while the bottom-up approach is often referred to as *norm emergence* and is considered in this section. Many researchers, such as [57, 69, 92, 116, 132], have spent some effort on norm emergence, including the emergence of a common language inside multiagent systems [70, 124], and the emergence of littering, smoking, or driving norms. This section introduces some of this work on norm emergence, providing a perspective on the subject relevant to the focus of this thesis.

2.8.1 Normative Games

Normative games had been widely used as environment in which various mechanisms that support norm emergence have been evaluated. In this section, we introduce the most important normative games that are used to study norm emergence in the literature.

Convention Emergence [121, 123, 142]: *Conventions* (or *conventional norms*) are a type of norm that naturally emerges with no threat of punishment. *Conventional norms* solve coordination problems, in which there is no conflict between the individual and the collective interests, since it is desired that everyone behaves in the same way, without consideration of which particular action is selected. Convention emergence is a process that starts with a population of agents with no initial preferences for any of the existing conventions. These agents must agree on a common convention (which is, by definition, only useful when all agents share the same one); once all agents have agreed on the same convention, the system has converged, and we can say that the norm has emerged. A classical example of convention emergence games is the iterated cooperation game (ICG) [61].

Language Games [70, 88, 109]: Providing a slightly different framework to classical convention emergence, Salazar et al. [109] propose a variation of the coordination game (as in *convention emergence*), implementing a *language coordination game* similar to [70]. In this game, agents must match words with concepts, creating a huge convention space.

The ultimatum game [111, 122]: In the ultimatum game, two agents play a game in which they interact with each other in order to share some money. The first agent proposes to the second how to share the money, and the second is free to accept or reject the proposal. If accepted, each agent takes its share, otherwise neither gets anything.

Emergence of Cooperation [141]: *Essential norms* are norms that solve or ease collective action problems, in which there is a conflict between the individual and the collective interests, making this the main difference with respect to *conventions*. Here, actions are not chosen randomly, because the targets' interests lie in the direction of action opposing observance of the norm, and the beneficiaries' interests lie in the direction of action favouring observance of the norm. *Essential norms* thus help to address situations in which individuals are tempted not to contribute to the common good. These problems are commonly known in the literature as *collective action problems*. The iterated prisoner's dilemma (IPD) [8] can be seen as a classic example of this.

Metanorm Games [7]: Punishments can be considered as a valuable means to direct a population toward a particular norm. However, the use of punishments is sometimes ineffective, especially when it is an altruistic punishment with no direct benefit to the party applying the punishment. Axelrod [7] introduced the key notion of metanorms to provide further encouragement for enforcing a norm. Here, those who observe a defection but do not punish the defectors are themselves punished, encouraging agents to punish norm defectors, and in turn persuading the defectors to desist from defecting. Norm compliance thus emerges among agents.

2.8.2 Learning

Various forms of learning have been an effective means for facilitating norm establishment among groups of self-interested agents. Learning by imitation has been used by various researchers [27, 54] in artificial intelligence, and by Epstein [36] in the context of norm emergence. In his model, agents must decide which side of a road to drive on, where the decision of each agent is determined by observation of which side of the road already has more agents driving on it, within a particular area. In this respect, agents imitate what the majority of their neighbours are doing. Borenstein et al. [18] study

the use of learning by imitation to enhance the evolutionary process through lifetime adaptation and show that imitation successfully achieve such a goal.

Similarly, Savarimuthu et al. [112] use imitation in considering the *ultimatum game* in the context of providing advice to agents on whether to change their norms in order to enhance performance. Here, each agent has a personal norm that defines its proposal strategy. In addition, agents are able to request advice regarding their proposal strategy from only one agent, the *leader*, which is believed to have the best performance in the requesting agent's neighbourhood. Moreover, agents are capable of accepting of refusing the advice according to their autonomy level.

Walker et al. [143] used a simple strategic update function in their model, based on Conte et al.'s [24] work. In their model, agents wander around searching for food in order to gain energy. However, since this movement causes them to lose energy, they need to find as much food as they can, and incur the least movement in doing so. For this reason, agents follow different strategies, and change from one strategy to another according to a *majority rule*, which instructs an agent to switch to another strategy if it finds that the other strategy is used by more agents that its current strategy. Shoham and Tennenholtz [119] proposed the use of the highest current reward (HCR) learning algorithm, by which agents learn to choose the action that brings the highest reward. HCR, which has also been used by Delgado et al. [33] in the context of a simple framework in which an agent makes a choice between two different actions and receives a positive payoff if they both choose the same action, or a negative payoff if their actions are different. Agents record the outcome of taking each of the two actions and pick the action with the better outcome for the next interaction.

A more complex form of learning has been used by Mukherjee et al. [90,117], who adopt Q-learning and some of its variants (WOLF-PHC and *fictitious play*) to show the effect of learning on norm emergence. They experimented with two different scenarios, first

with homogeneous learning agents (where all agents have the same learning algorithm), and second with heterogeneous learning agents (where agents can have different learning algorithms). Their results suggest that norm emergence is achieved in both situations, but is slower in heterogeneous environments.

2.8.3 Topologies

As this suggests, there has been much work that focusses on the issue of norm emergence in societies of interacting agents. However, most of this concentrates on analysing norm emergence over fully-connected networks [17, 117, 120, 143], and it was only relatively recently that attention shifted towards the effect of the structure of these societies. In this section, we review that part of the literature that does address these concerns.

The earliest work on examining the impact of topologies on convention emergence belongs to Kittock [64], who studied the effect of regular lattices on the iterated cooperation game (ICG) [61] and the iterated prisoner's dilemma (IPD) [8]. Both are iterative games in which two agents choose one of two actions with their reward depending on the combination of their choices. Agents choose actions based on the highest current reward (HCR) learning algorithm. The results obtained suggest that different convergence rates are observed with different topologies and, in particular, that a longer network diameter (being the longest path between any two nodes) make it more difficult for the convergence to arise.

Urbano et al. [133] use a classical convention emergence game to study the influence of interaction topologies (random, regular, small world and scale free topologies) on the External Majority strategy update rule (EM) originally proposed by Shoham and Tennenholtz [118]. In particular, they investigate the effect of different memory sizes on the rate of convergence. Their empirical results show that a memory size of 3 is

the best option for all types of topologies. More recently, Franks et al. [41] have shown that inserting a small amount of influencer agents — those with specific conventions and strategies — is enough to manipulate the convention adopted by large societies. However, their results depend on the underlying network, with scale-free networks easing the emergence of conventions in comparison to small worlds, over which convention emergence is slower and with higher cost.

In a particular effort, Delgado et al. [32,33] study the emergence of coordination in scale-free networks. Their study involves an interaction model of a multi-agent system, by which they try to analyse how fast coordination can spread among agents. Coordination here is represented through agents being in the same state, which is achieved when 90% of the agents do so. The results of the work demonstrate that coordination can indeed be achieved over scale-free networks, but in a rather restricted setting. Similarly, Sen et al. [115] use a game to investigate norm emergence over lattices and scale-free networks. In particular, they analyse the effect of increasing the number of actions available to agents, as well as the effect, on the speed of norm emergence, and of increasing the number of agents in both scale-free networks and lattices. Their results suggest that both increasing the number of actions *and* increasing the number of agents causes a delay to norm emergence in the population over a scale-free network. Similarly, norm emergence in lattices is much slower when agents have a larger set of actions to choose from, or when the number of agents in the population is increased. Overall, their analysis shows that, for a small set of actions, it is faster for a norm to spread in a ring than in other topologies, followed by fully connected structures, and then scale-free networks. In contrast, for a large set of actions, it turns out that this is much faster in scale-free networks than in rings and fully connected structures.

The models used in these previous pieces of work are relatively unsophisticated, with only two agents are involved in a single interaction, and reward values remaining fixed

and not changing during the game. In response, Villatoro et al. [140] adopted the same concept of two-agent interactions, but introduced the notion of the reward of an action being determined through the use of the memory of agents, thus adding some dynamism to the model. Here, the reward of a certain action is determined by whether the action represents the majority action in both agents' memories, and the reward is proportional to the number of occurrences of this majority action in the their memories. However, it is not clear from where these rewards derive nor who applies them, as agents only have access to their memory. With regard to the interaction network, their work illustrates that increasing the neighbourhood size of a lattice accelerates norm emergence. In contrast, in the case of scale-free networks, norms do not emerge using the basic model. This is because of the development of *sub-conventions* that are persistent and hard to break, and which prevent the whole population from converging towards a single convention. A solution to this problem was found by giving *hub* agents (those with the majority of connections to others) more influence on the reward function.

An interesting property that has been explored with regard to networks is that of community structure [99,100], which has been shown to exist in many real-world social networks [94]. If the nodes of a graph form a set of groups that are themselves densely connected, but loosely connected with other groups, then such a graph is said to have the community structure property. O'Riordan et al. have shown that given a strong community structure, robust cooperation emerges among a population of agents that are playing the N-player prisoner dilemma [28]. This has been shown to be effective over both lattice [99] and small world networks [100].

Savarimuthu et al. [112] analyse the effect of *advice* on norm emergence over random and scale-free networks. For this reason, they use the *ultimatum game* and their results show that norm emergence increases in speed over both random and scale-free networks

with an increase in the average degree of connectivity. Furthermore, they have shown [110] that their model works over dynamic network topologies that are generated using Gonzalez's model [45]. More recently, Mungovan et al. [91] introduced the idea of weighted random interaction by which agents are able to interact with random members of the population based on the distance, so the closer an agent is to another, the more likely there will be an interaction between these two agents. Their results suggest that dynamic interaction helps in easing emergence especially in breaking local biases that are normally hard to break.

2.8.4 Punishment

Even though Axelrod's simulation model was limited, he provided a valuable explanation for the emergence of cooperation and the stability of punishment. Since then, however, many others [37, 56, 68, 95, 134] have been concerned with the evolution of altruistic punishment, and some researchers in particular have empirically shown that the existence of punishment allows for the emergence and stability of cooperative strategies within human populations. Indeed, in the last decade, an important body of work concerned with multi-agent systems and punishment has developed, analysing all aspects related to the regulation of normative behaviour [51].

Prior work in this area has mainly addressed the use of different forms of punishment in order to obtain the desired system behaviour. For example, de Pinninck [30] takes the use of reputation to its most extreme, by allowing agents to definitively remove interactions with norm-violators. Here, ostracism leads to satisfactory results in the presented peer-to-peer example, but this approach suffers from the weakness that the norm-violators lose all possibility of interaction, and are not allowed to adapt and alter their behaviour after punishment. Savarimuthu et al. [113] shows that peer-to-peer

punishment is effective in achieving norm emergence in virtual on-line societies when the cost of punishing is low.

In the context of applying punishment to alter the behaviour of the punished agent, Villatoro [139] introduces a simple heuristic for adaptive punishment in a prisoner's dilemma setting. This adaptive punishment approach obtains good results but involves a longer adaptation time. Andrighetto and Villatoro [3] have also studied the effect of sanctions, which they consider to be different from punishments in the sence that sanctions carry extra information to the violator, emphasising the mistake it has made by violating the norm. They have shown that sanctions can achieve very similar results to punishments with lower cost, and can make a norm more stable when the enforcement mechanism is interrupted.

2.9 Conclusion

The focus of this thesis is on distributed large scale systems such as peer-to-peer (P2P) file sharing, in which there is a need to regulate the behaviour of system participants to avoid malicious actions and achieve overall social order. Such systems avoid using any form of central control due to the magnified expense associated with the use of such techniques. Because of this, emergence plays a big role in achieving such goals in these systems, since it allows participants to regulate each others' behaviour through the interactions in which they engage. This is usually achieved through some kind of incentivisation that agents provide for each other, typically through punishment of those who act maliciously or reward for those who do not. In fact, punishment is the most common form of such incentivisation that has been used in the literature on norm emergence. In addition, the connections between the participants of these systems follow certain patterns or topologies. For example, P2P file sharing systems

are known to form a structure that follows the scale-free topology, adding complexity to the ability to establish emergence.

As described in Section 2.8.1, the results obtained with Axelrod's model suggest that metanorms can offer an effective mechanism to achieve emergence in distributed large scale systems. However, the model itself has limitations: it has limited analyses of the obtained results; it assumes a central control authority; and it does not consider network topologies.

Moreover, as mentioned earlier, punishment is a common means of encouraging emergence. However, models that involve punishment tend to assume static punishment values that do not change according to the context. This is a rather limited view, since it may not be always possible to determine a suitable static punishment that allows a norm to emerge without the use of excessive punishment. Less punishment may not allow the norm to emerge, while more punishment that is unnecessary might overly constrain the behaviour of participants or even hinder their participation at all.

In consequence of this discussion, the work in the remainder of this thesis first undertakes a deeper analysis of Axelrod's model in order to precisely identify its strengths and weaknesses. Then, based on this analysis and the concept of metanorms, we develop a model that is independent from any central control, and that incorporates the influence of topologies. Using this as a platform, an investigation into developing an adaptive punishment mechanism is described, and the developed techniques integrated, to provide a novel and effective approach to norm emergence with metanorms and topological constraints.

Chapter 3

An Analysis of Norm Emergence in Axelrod's Model

3.1 Introduction

As we have seen, and as has been suggested by many (e.g., [15, 16, 34, 44, 150]), *norms* provide a valuable mechanism for regulating or constraining human societies. Perhaps the most obvious and clear manifestation of norms is when they arise through the explicit introduction of laws that are established by legislatures, for example, or through rules or bye-laws of smaller groups such as member clubs. However, norms are also valuable when there is no central authority, and they emerge as a result of individual behaviour, in order to establish some coherence or stability in a group. It is this latter aspect that has been the focus of several researchers (e.g., [36, 40, 113, 140, 152]) perhaps most notably Axelrod, whose seminal paper in 1986 offered a model of norms and metanorms [7] that has since been slightly investigated [42, 103].

Axelrod's model is a game in which different agents decide whether to defect or cooperate (comply). Agents may also observe others and have the ability to punish those who defect. An agent's behaviour is assessed by means of a careful scoring system that simulates the potential rewards and penalties associated with norm violation and enforcement. The agent population evolves through a number of iterations, with a mechanism whereby successful behaviour (as measured by the scoring system) tends to be replicated and unsuccessful behaviour tends to be discarded. In each iteration, each replicated behaviour is subjected to a small chance of mutation, reflecting the feature that an agent may occasionally change its strategy, irrespective of past habits. The strategy of each agent in determining whether to defect and whether to punish others is based on two different attributes, *boldness* (encouraging agents to defect) and *vengefulness* (encouraging them to punish others), which are distinct for each agent. The idea is that a system eventually resulting in all agents having high vengefulness and low boldness corresponds to norm emergence, since they will punish defection but they will not themselves defect. Key to Axelrod's model is the notion of *metanorms*, secondary norms that help to enforce compliance with primary norms by punishing agents that do not themselves punish a defector. By using metanorms, Axelrod was able to establish norms in his experiments.

However, as was more recently shown by Galan and Izquierdo [42], Axelrod's results are dependent on both certain assumptions and some very specific and arbitrary conditions. In this chapter, we elaborate on the work of Galan and Izquierdo, showing that their results, too, rely on some assumptions and conditions. We also provide a further analysis of Axelrod's model, drawing out some important considerations for the establishment of norms more generally.

The rest of this chapter is organised as follows, Section 3.2 provides a description of Axelrod's model, followed, in Section 3.3, by a more detailed analysis of the results

than provided elsewhere. Then, in Section 3.4, the duration of the game, a critical part of Galan and Izquierdo's analysis, is reviewed, with different results, leading to a new consideration of the circumstances for norm collapse (when norms are not established). In Section 3.5, the impact of the reproduction policy is analysed, and in Section 3.6, a consideration of the impact of mutation is provided. The chapter concludes in Section 3.7 with a discussion and conclusions on the significance of the obtained results.

3.2 Axelrod's Model

Axelrod's Model has been introduced in two stages detailed next.

3.2.1 The Norms Game

TABLE 3.1: The Norms Game terms (from [7])

Term	**Description**	**Value**
i, j	Individuals	A number to identify individual agents
S	Probability of a defection being seen by any given individual	Uniform distribution from 0 to 1
B_i	Boldness of i	Uniform distribution from 0 to 1
V_i	Vengefulness of i	Uniform distribution from 0 to 1
T	Player's temptation to defect	$+3$
H	Hurt suffered by others as a result of an agent's defection	-1
P	Cost of being punished	-9
E	Enforcement cost, i.e. cost of applying punishment	-2

Axelrod's *norms game* adopts an evolutionary approach in which successful strategies are multiplied over generations, potentially leading to convergence on norms. Each individual, or agent, can choose to *defect* by violating a norm, and such behaviour has

a particular known chance of being observed or *seen* (S). An agent i has two decisions, or strategy dimensions, as follows. First, it must decide whether to defect, determined by its *boldness* (B_i). Second, if it sees another agent defect (determined by S), it must decide whether to punish this defecting agent, determined by its *vengefulness* (V_i), which is the probability of doing so. (A full list of the terms used in this experiment is provided in Table 3.1). If $S < B_i$, then agent i defects, receiving a *temptation payoff*, $T = 3$, while *hurting* all the others with payoff $H = -1$. If a defector is *punished* (P), the payoff to the defector is $P = -9$, while the punishing agent pays an *enforcement cost* $E = -2$. The initial values of B_i and V_i are chosen at random from a uniform distribution of a range of eight values between $\frac{0}{7}$ and $\frac{7}{7}$.

Axelrod's simulation has a population of 20 agents, with each agent having four opportunities to defect, and the chance of being seen for each drawn from a uniform distribution between 0 and 1. After playing a full round (all four opportunities), scores for each agent are calculated in order to produce a new generation, as follows. Agents that score better or equal to the average population score plus one standard deviation are reproduced twice in the new generation. Agents that score one standard deviation under the average population score are not reproduced at all, and all others are reproduced once. Although this may produce a new generation with a different number of agents, Axelrod maintains the number of agents at 20 over subsequent generations, but does not specify how. Finally, a mutation operator is used to enable new strategies to arise. Since B_i and V_i (which determine agent behaviour) take 8 possible values, they need three bits to be represented. Mutation is thus applied by flipping one of these bits whenever an agent is reproduced, with a 1% chance.

Axelrod's experiment has five runs, each with 100 generations. The final results of these runs are as follows: two runs achieved high average boldness and almost zero average vengefulness indicating no norm emergence at all; two other runs achieved low average

boldness and vengefulness; and only the final run achieved a high level of average vengefulness and very low average boldness, indicating a partial establishment of a norm against defection. Axelrod introduces a metanorm model, in which an additional mechanism is considered to support better norm establishment.

3.2.2 The Metanorms Game

The key idea underlying Axelrod's metanorm mechanism is that some further encouragement for enforcing a norm is needed. This is accomplished by introducing a *metanorm* for punishment of those who observe a defection but do not punish the defectors. In this new metanorms game, if an agent sees a defection but does not punish it, this is considered as a different type of defection, and others in turn may observe this defection (with probability S) and apply a punishment to the non-enforcing agent. However, if agents decide not to apply this second level punishment, there is no risk of another level of punishment being applied to them. As before, the decision to punish is based on vengefulness, and brings the defector a punishment cost of $P' = -9$ and the punisher an enforcement cost of $E' = -2$ (see Table 3.2 for the list of terms that used in the metanorm game). Applying this new metanorm game to the same simulation as before gives runs with high vengefulness and low boldness, which is exactly the kind of behaviour needed to support establishment of a norm against defection.

3.3 Analysis of Axelrod's Model

In seeking to replicate Axelrod's results, it becomes clear that in order to reimplement the model, some assumptions need to be made, about which Axelrod says nothing. First, the model does not specify how the constant population level is maintained after reproduction, when there are three possible scenarios. *(i)* The new population is smaller

TABLE 3.2: Metanorms Game terms (from [7])

Term	**Description**	**Value**
i, j	Individuals	A number to identify individual agents
S	Probability of a defection being seen by any given individual	Uniform distribution from 0 to 1
B_i	Boldness of i	Uniform distribution from 0 to 1
V_i	Vengefulness of i	Uniform distribution from 0 to 1
T	Player's temptation to defect	$+3$
H	Hurt suffered by others as a result of an agent's defection	-1
P	Cost of being punished	-9
E	Enforcement cost, i.e. cost of applying punishment	-2
P'	Cost of being punished for not punishing a defection	-9
E'	Cost of punishing someone for not punishing a defection	-2

than the original. In our re-implementation, additional agents are randomly selected from the resulting population and replicated. *(ii)* The new population is equal in size to the original, in which case nothing new is needed. *(iii)* The new population is larger than the original. In this case, our position is to select the required number of agents at random from the relevant set for reproduction. Second, we assume the score of each agent is set to 0 at the beginning of each generation.

Axelrod's experiments are repeated running the norms game 10 times, with the results being shown in Figure 3.1, where the diamonds represent the value of the mean average boldness and vengefulness of the final generation's population. As can be concluded from the figure, the results obtained are similar to those of Axelrod, with one run having high vengefulness and low boldness, two runs with exactly the opposite (high boldness and low vengefulness), and all other runs with low values for both boldness and vengefulness.

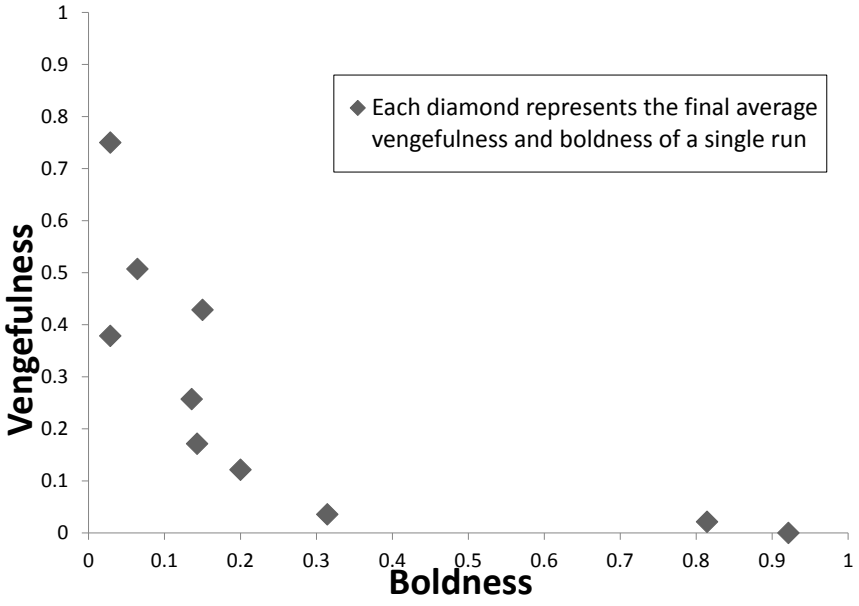


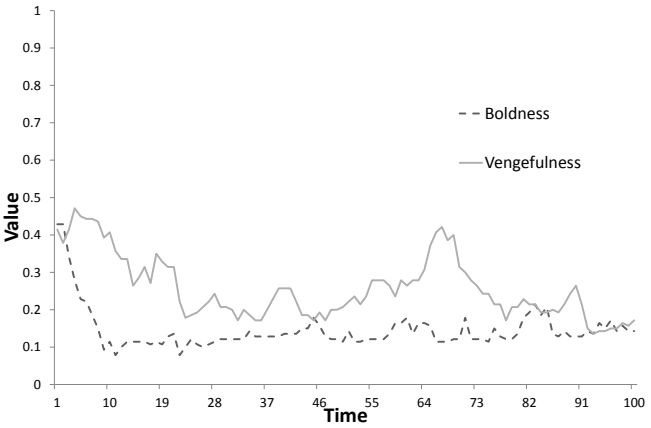
FIGURE 3.1: Norms game overall results

In order to establish how these results arise, changes to boldness and vengefulness for each individual were monitored. Figure 3.2 provides some sample graphs illustrating such changes, showing the average boldness and vengefulness as they vary over generations (and hence time). In particular, Figure 3.2(a) shows one run ending in the most common result, low boldness and low vengefulness. The run starts with average boldness and vengefulness of about 0.5 (as initial values for B_i and V_i are taken from a uniform distribution over $\{\frac{0}{7}, \ldots, \frac{7}{7}\}$). In the early stages, boldness decreases slightly, indicating that individuals with higher boldness are eliminated. This is because high boldness causes an agent to defect, yet defecting with average vengefulness can be costly, as the agent is likely to be punished, leading to a low score. Subsequently boldness stabilises at a low level. With a low average boldness in the population, being vengeful becomes costly, causing agents with low vengefulness to get favoured over those with high vengefulness, with the latter getting eliminated when reforming the new generation. Finally, the values stabilise at particular low values for both boldness

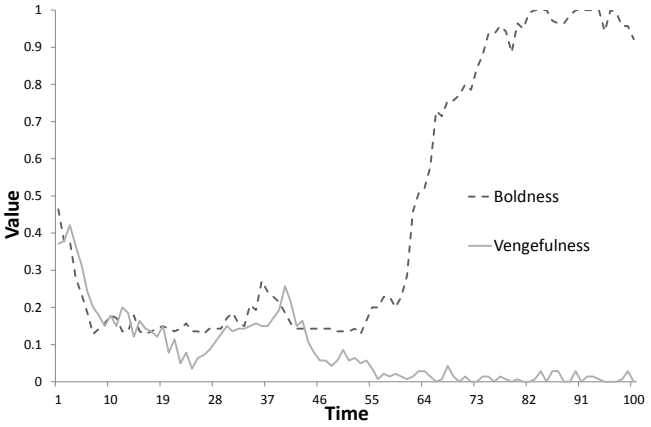
and vengefulness until the end of the run.

In the cases that result in high boldness and low vengefulness (an example run is shown in Figure 3.2(b)), the run starts as before, with both values reducing. However, around the 60th generation, the value of boldness increases sharply until it reaches 1, where it remains. This can be explained by a dramatic change to one individual's boldness, due to mutation, in an agent population with particularly low values of vengefulness. In turn, this facilitates the individual's survival, dominating the others and allowing it to propagate its high boldness across the population. Here, a high score is attained by defecting without punishment (due to low vengefulness), which also hurts others and lowers their scores. In the final case, as shown in Figure 3.2(c), the run ends with high vengefulness and very low boldness: while eliminating all those with high boldness, only individuals with high vengefulness remain, so there are no individuals with low boldness and low vengefulness, and those with high vengefulness and low boldness survive and dominate.

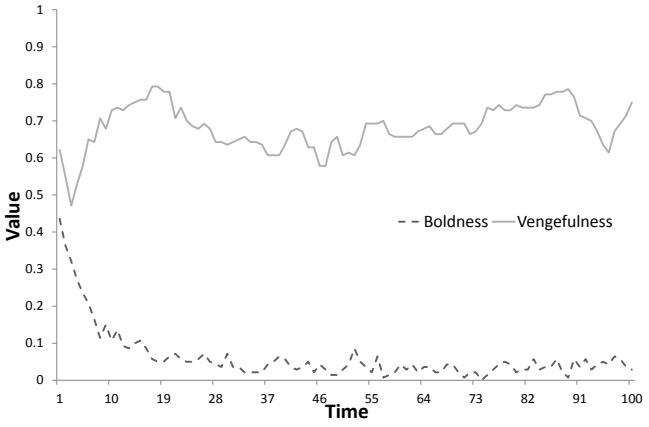
By introducing metanorms, Axelrod aimed to address the problems identified above. Our own simulation concerning the metanorms game (see Figure 3.3) also provides similar results to Axelrod. Again, a deeper analysis of the results is conducted. As shown in Figure 3.4: in the metanorms game, the population starts eliminating high boldness individuals as before, but now also eliminates low vengefulness individuals. The latter trend is due to the introduction of the metanorm according to which failure to penalise a defector may also be penalised. This results in a population with high vengefulness and low boldness, which survives until the end of the run.



(a) Norms game: low boldness and low vengefulness



(b) Norms game: high boldness and low vengefulness



(c) Norms game: low boldness and high vengefulness

FIGURE 3.2: Norms game: analysis of runs

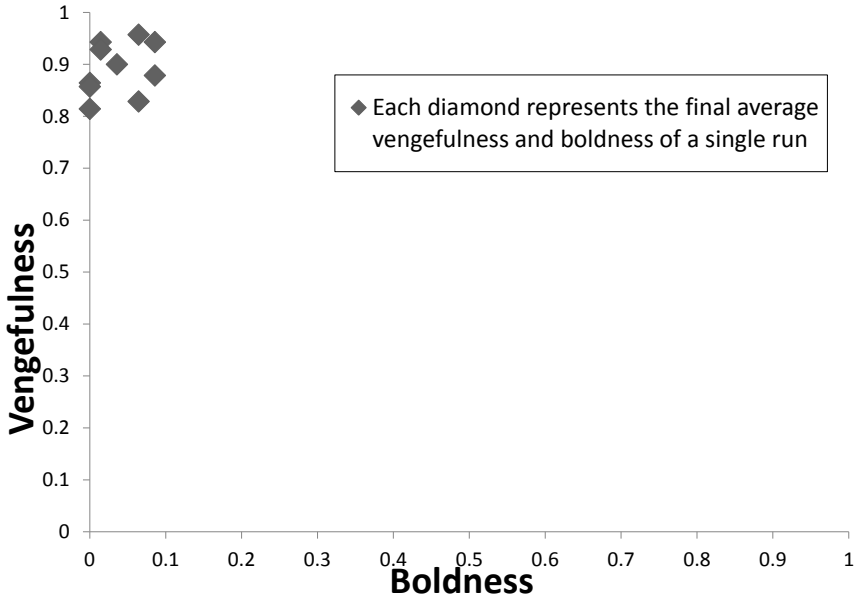


FIGURE 3.3: Metanorms game overall results

3.4 Game Duration

To provide a stronger analysis of Axelrod's model, the experiments are repeated over a *long duration*, 1,000,000 generations (and 10 runs), as opposed to 100 generations (in Axelrod's experiments and our experiments of Section 3.3). In our norms game simulation, the game starts with boldness decreasing, and then vengefulness decreasing until they both settle at a low level, which is consistent with Axelrod's results. However, an agent with high boldness can be introduced to such a population through mutation, and would dominate others since it is not punished due to low levels of vengefulness. Clearly, running the experiment for a longer period increases the possibilities for this to occur and, as shown in Figure 3.5, this always leads to norm collapse.

In undertaking their own analysis of Axelrod's metanorms game, Galan and Izquierdo [42] increased the number of generations in a run and found different results. By including 1,000,000 generations, in 1,000 runs, nearly 70% ended in *norm collapse*, as opposed

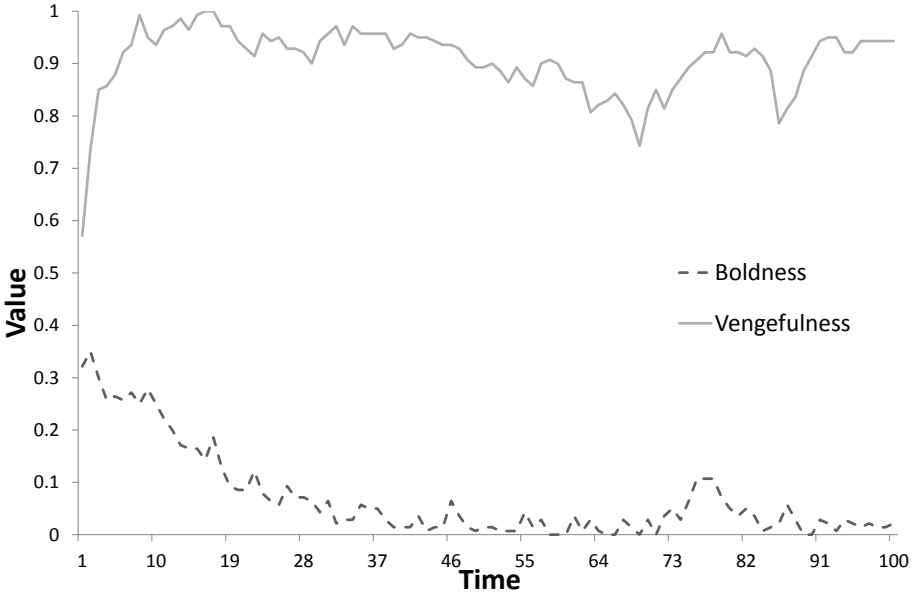


FIGURE 3.4: Metanorms game analysis

to Axelrod's *norm establishment*. According to Galan and Izquierdo, this is because vengefulness is costly in a population in which violation is rare. Thus, agents with low vengefulness are favoured over agents with high vengefulness, leading to a significant decrease in vengefulness, encouraging defection, and in turn causing boldness to increase. The results of Galan and Izquierdo suggest that metanorms are not as useful as it might seem from Axelrod's results.

By analysing these cases to determine the reasons for these results, it is clear that the runs begin in the same way as previously observed, by eliminating individuals with high boldness and low vengefulness, stabilising on those with high vengefulness and low boldness. Then, however, mutation causes vengefulness to reduce. If an agent x with high vengefulness and low boldness changes through mutation to give lower vengefulness, while boldness for all remains low, there is no defection and the mutated agent survives. In addition, if boldness then mutates to become just a little higher for a different agent y, with average vengefulness remaining high, x will still rarely defect because of relatively low boldness. If it *does* defect, and *is* seen by others, it receives

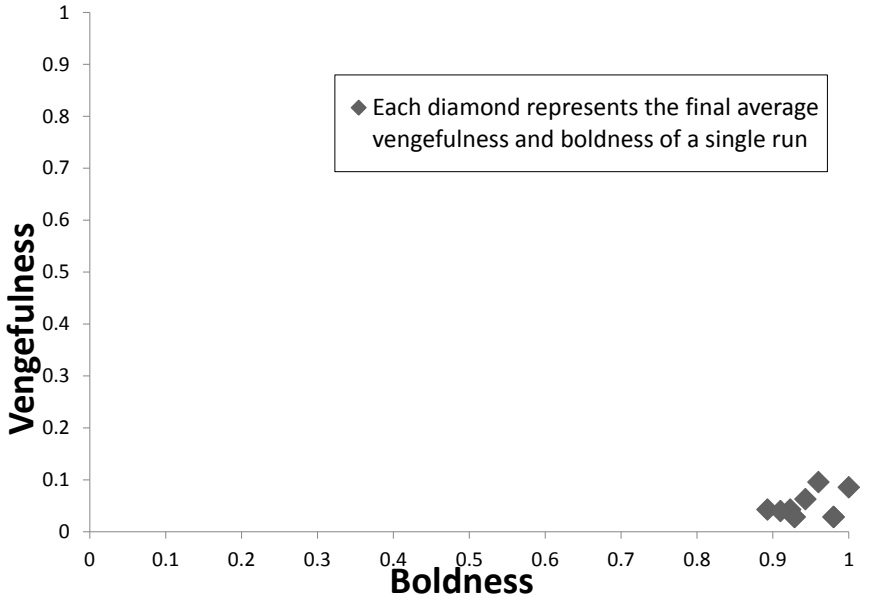


FIGURE 3.5: Norms game for 1,000,000 generations

a low score, unless it is not punished, in which case the non-punishing agents may themselves be punished because of the high vengefulness in the general population. Here, agent x may not punish y either because of the low probability of being seen (which must be below the low boldness level to have caused a defection) or because it has mutated to have lower vengefulness. In the former case, x will not be punished for non-punishment since it has not observed y's defection, but in the latter case, x might be punished if it is seen by others. However, we know that the probability of being seen is low because agent y has defected (and $S < B$ for defection to take place). In this case, y is eliminated, while x remains, because the likelihood of y's defection being seen by just one agent is relatively high, but the likelihood of agent x's non-punishment being seen requires first y's defection being seen by x, and then x's non-punishment being seen by others, the combination of these being extremely unlikely.

If the values of vengefulness continue to decrease in this way, the population can arrive at a situation with very low average boldness and vengefulness. At this point, a single

mutation to boldness could then cause the mutant to dominate the others due to the general lack of vengefulness in the population. The key question here is why, in cases of high boldness and low vengefulness, mutation of vengefulness from a very low value to a significantly higher value does not cause boldness to decrease. Here, such a mutant should punish all others for defecting *and* for not punishing defectors. However, these punishments also incur significant enforcement costs, all of which are borne by the punishing agent, potentially exceeding the penalty meted out to the defectors and those agents who fail to punish others.

This analysis suggests that both mutation and sanctioning costs play a major role in collapsing norms. However, there is an additional factor that gives rise to these results, a particular characteristic of the underlying model which, in certain circumstances, and with very subtle change, can give a very different outcome. We consider this next.

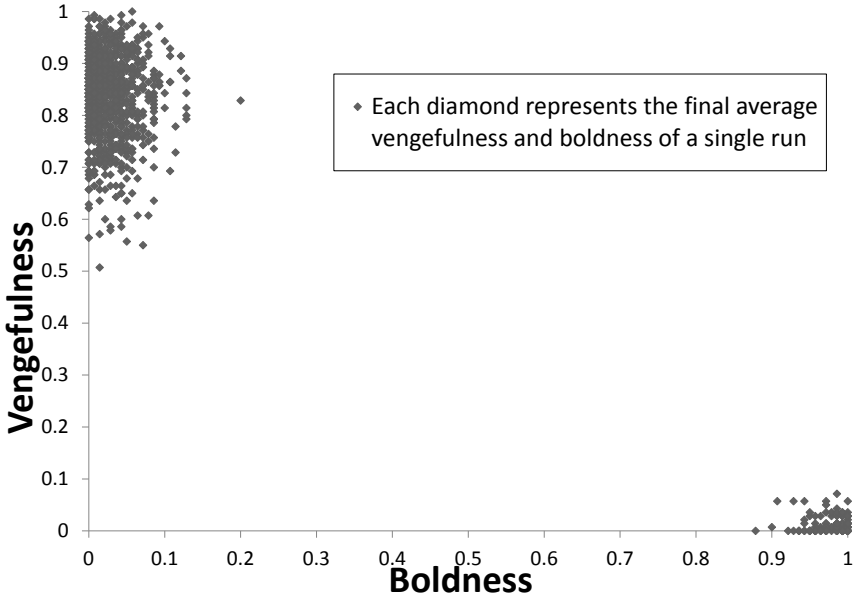


FIGURE 3.6: Metanorms game for 1,000,000 generations

3.5 Reproduction and Norm Collapse

As specified earlier, a run of the metanorm game settles at very low boldness and very high vengefulness at a certain point. For this to change to the opposite situation of low vengefulness and high boldness, a sequence of modifications that lower vengefulness must occur, and another sequence of modifications that increase boldness must also occur.

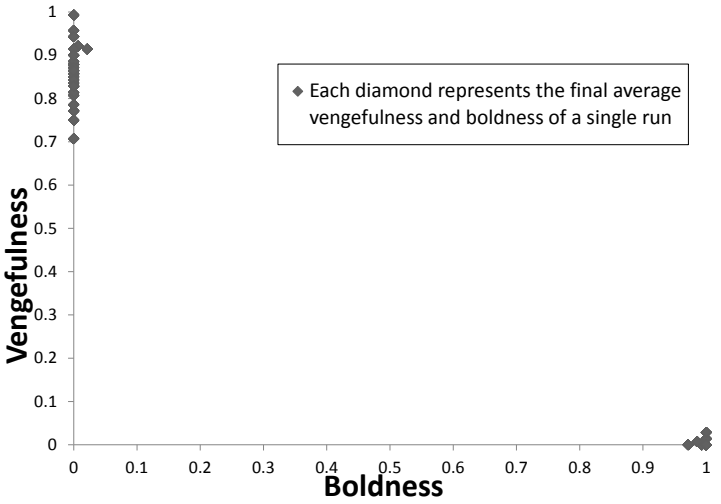
Another factor that could contribute to norm collapse, which is not considered above, is the reproduction policy over generations, especially where the boldness of the entire agent population is very low. Specifically, when all individuals have boldness around 0, defection rarely happens, and their scores (which change only when agents enforce, are hurt, or are punished) are 0. As a result, the average score and standard deviation are also 0, so that all agents have a score equal to the average score plus one standard deviation. According to Axelrod's rules, agents in this situation should be replicated *twice* when forming the new generation. However, duplicating the individuals in this case does not seem sensible since it does not fulfil the original purpose suggested by Axelrod, of giving individuals with better scores a greater chance of survival. Hence, to study the effect of the reproduction policy, we conducted simulations in which agents are only replicated once in the above special case (as apposed to Galan and Izquierdo where all agents in such case are duplicated). The results obtained of running the metanorm game with many more generations are similar to those of Galan and Izquierdo, but with a different proportion giving rise to defection. As shown in Figure 3.6, 128 out of 1,000 runs (or 13%) of 1,000,000 generations ended in norm collapse, as opposed to 70% reported by Galan and Izquierdo.

The reason behind the different results is that replicating an entire population of non-defecting agents increases the likelihood of significant fluctuations in vengefulness over

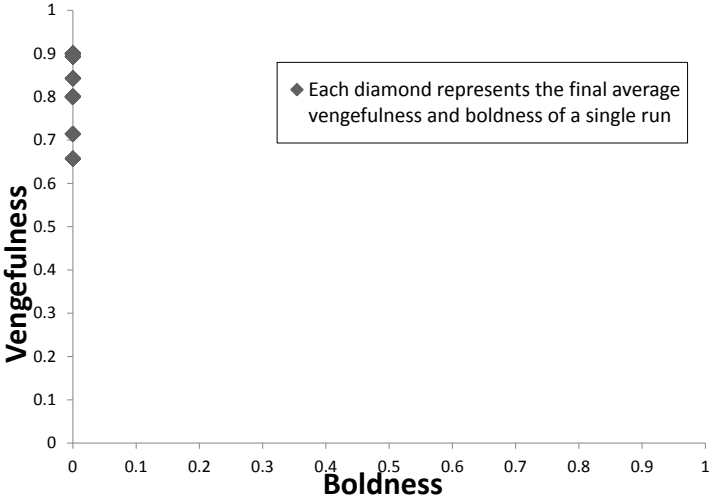
subsequent generations. For example, in one phase of a run using the approach of Galan and Izquierdo in which all agents have 0 boldness, five agents have vengefulness of 0, eleven with 1 and four with 0.8, the next generation includes eight agents with vengefulness of 0, seven agents with 1, and five agents with 0.8, simply due to the replication policy. This means that average vengefulness drops from 0.71 to 0.55 and, as boldness continues at 0, replication again makes this worse. However, note that replication could cause the opposite, increasing the number of agents with high vengefulness over those with low vengefulness.

As new generations of agents with low boldness are evolved for more iterations, it becomes more likely to observe the following combination of events. First, the levels of vengefulness decrease repeatedly through a sequence of downward fluctuations until they reach very low levels. Then, until vengefulness levels fluctuate back upwards, this creates a temporarily fertile environment for defectors. Next, the boldness of one or a few agents increases to a high level due to mutation. This causes the bold agents to defect and be replicated in subsequent iterations of the game. The end result of the phenomenon is an agent population where high boldness and defection are so prevalent that being vengeful leads to extinction. Thus, the game reaches a stable situation where norm collapse is ingrained in the population (i.e. low vengefulness and high boldness).

This end state is reached in a proportion of experimental runs of both Galan and Izquierdo's experiments and our own experiments because it is more likely to reach the required preconditions when repeating the experiment for 1,000,000 generations (i.e. over a long duration). However, a much larger proportion of Galan and Izquierdo's runs end in stable norm collapse because their replication policy allows for much more significant fluctuations of vengefulness.



(a) 1,000,000 generations and 0.001 mutation rate



(b) 1,000,000 generations and 0.0001 mutation rate

FIGURE 3.7: Metanorms game with different mutation rates

3.6 Mutation

As discussed above, mutation is significant in determining when norm collapse occurs. Galan and Izquierdo [42] argue that decreasing the mutation rate from 0.01 to 0.001

allows norm collapse to arise much earlier. They present an example in which a mutation rate of 0.001 allows the population to converge toward norm collapse in about 25,000 generations as opposed to 300,000 with 0.01. However, they do not explain the reasons, and do not explore the more general effect of mutation rate on norm collapse or establishment.

In seeking to consider this further, we undertook an experiment consisting of 1000 runs of 1,000,000 generations, using the mutation rate of 0.001 suggested by Galan and Izquierdo, with results shown in Figure 3.7(a). Clearly, these results support Galan and Izquierdo's argument: 40% of the runs ended in norm collapse. However, decreasing the mutation rate further does not have the same effect. In particular, a mutation rate of 0.0001 resulted in all runs ending in norm establishment, shown in Figure 3.7(b).

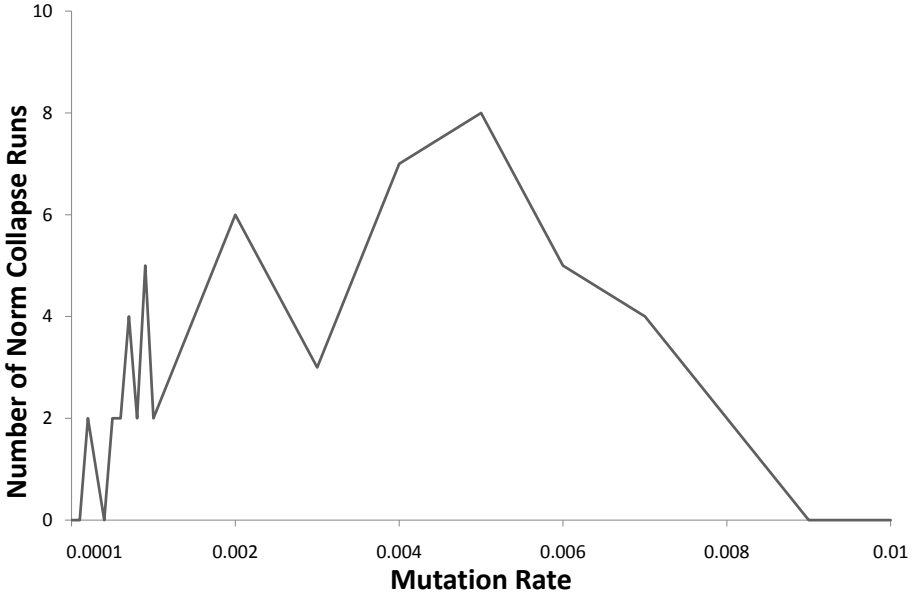


FIGURE 3.8: Metanorms game: 1m generations; 0.0001–0.01 mutation rate

The relation between mutation rate and norm collapse is thus unclear. To understand this better, we performed a further series of experiments. Figure 3.8 illustrates the result of different experiments that consists of 10 runs each, with a range of mutation rates between 0.0001 and 0.01. As can be observed from Figure 3.8, the mutation rate

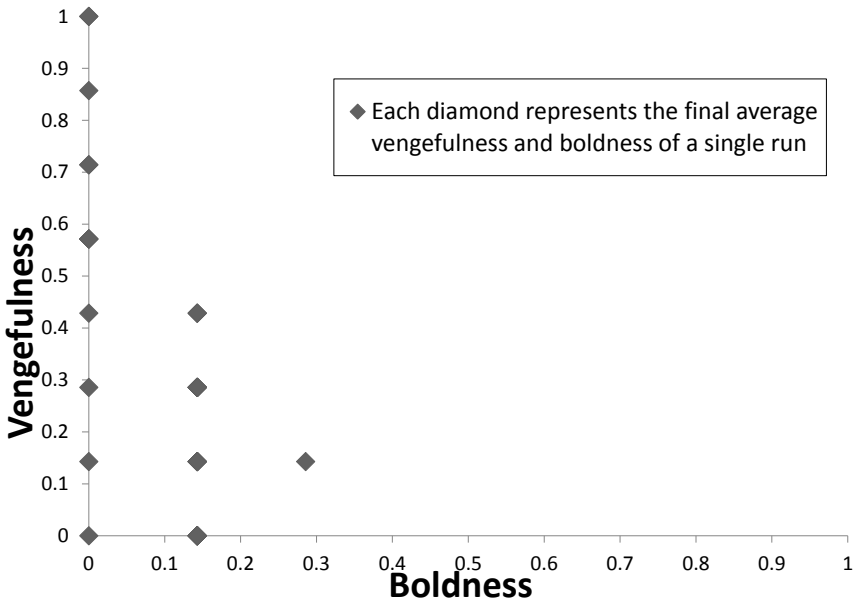
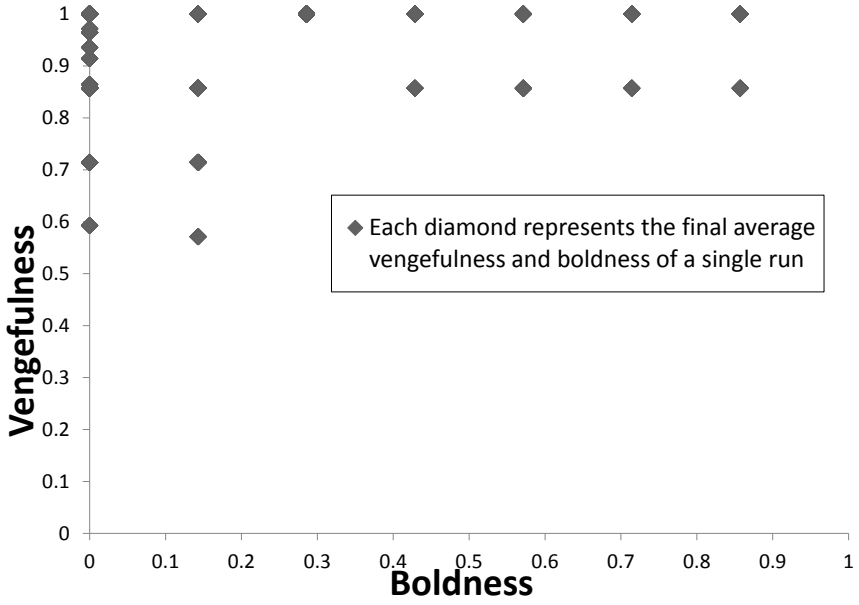


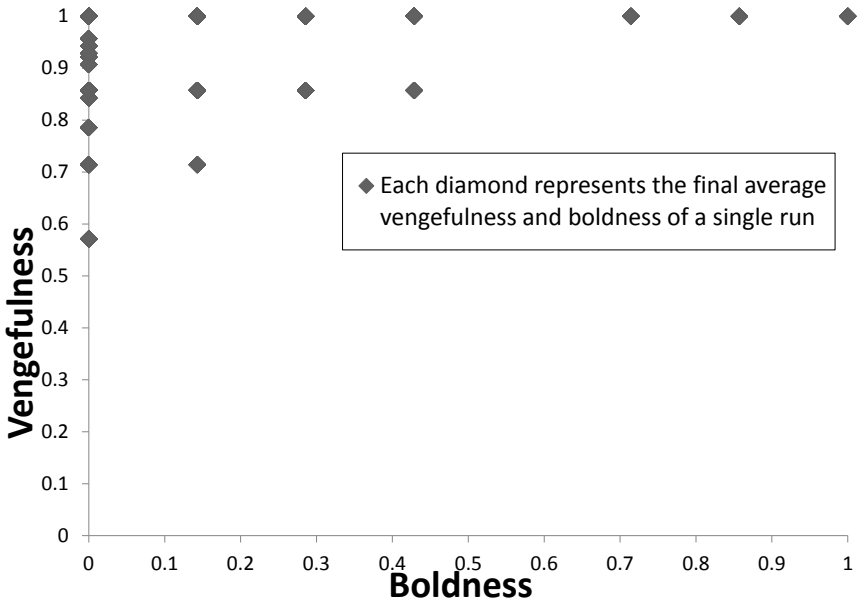
FIGURE 3.9: No mutation norms game: 1,000,000 generations

seems to play an important role in causing norms to collapse. Decreasing the mutation rate below 0.01 has a major effect on the proportion of runs ending in norm collapse, with a peak around mutation rate values of 0.005 giving norm collapse in 80% of runs. However, decreasing the mutation rate further causes the proportion of runs ending in norm collapse to drop back (with fluctuations) until it reaches 0 with a mutation rate of 0.0001. While these results suggest a potentially interesting relationship, further work is needed to establish the exact correlation, which is out of the focus of this thesis.

Nevertheless, we can say that given these results, removing mutation should avoid norm collapse. In the norms game, after the population stabilises at a low level of both vengefulness and boldness, mutation of an agent's boldness from low to high allows it to dominate, and as a result eliminate others, which leads to norm collapse (as previously shown in Figure 3.5). Figure 3.9 illustrates the result of a no mutation norms game that consists of 1000 runs, with 1,000,000 generations each. As expected, removing mutation avoids norm collapse and leaves the population with the other two situations.



(a) 100 generations without mutation



(b) 1,000,000 generations without mutation

FIGURE 3.10: No mutation metanorms game

In the metanorms game, as we have seen, mutation seems to have a great effect on moving away from norm establishment. By removing mutation, we might expect to guarantee norm establishment. To corroborate this, we performed two experiments for two different durations, with results shown in Figure 3.10(a) for 100 generations and 1000 runs and Figure 3.10(b) for 1,000,000 generations and 1000 runs. Surprisingly, the result was not as expected. A high level of vengefulness is maintained in almost all the runs, but a high level of boldness is also observed in some, and hence a high level of defection in the population, despite the associated punishment. This is because the final result of each run primarily depends on the initial distribution of vengefulness and boldness: individuals with high vengefulness and high boldness at the start are favoured over those with average vengefulness and low boldness. As a result, they survive and dominate the others. More importantly, the final result is determined within the very first generation so that running the experiment for longer has no impact and there is no change to the population once the levels of vengefulness and boldness stabilise.

3.7 Discussion and Conclusions

It is clear that Axelrod's model exhibits many interesting aspects, and relies on characteristics that provide different results with different assumptions or instantiations. We have explored several of these in relation to our experiments, found some distinct features and, as a result, we can also provide a characterisation of the nature of norm establishment or collapse more generally.

Given the analysis through the chapter, it should be clear that norm establishment lies in the region where vengefulness is high and boldness is low; similarly, norm collapse lies in the region where vengefulness is low and boldness is high. This is as used by

Axelrod in his model, and underlies the aim of the initial experiments. It is illustrated graphically in Figure 3.11. However, we can also characterise the region where vengefulness is low and boldness is low as tending to norm collapse: this is a region of benign behaviour since boldness is low and defection is unlikely, but it is unstable since a mutation to boldness may take it higher, leaving vengefulness low, and causing norm collapse. Conversely, the region where both vengefulness and boldness are high is tending to norm establishment: it is undesirable since boldness is high and there are many defections, but these defections are likely to be punished. If vengefulness does cause punishment, then it is likely that boldness will drop, leading to norm establishment. Given this view, we can reinterpret the previous experiments. For example, it is clear that the run shown in Figure 3.2(a) ends in a state tending to norm collapse, the run in Figure 3.2(b) ends in norm collapse, and the run in Figure 3.2(c) ends in norm establishment, just as the run in Figure 3.3.

While we have analysed the impact of duration, reproduction policy and mutation rate on Axelrod's game, some of the results, such as the exact relation of mutation rate to norm collapse also require further analysis, which is not of the focus of this thesis. For example, some results in the chapter suggest that the sanctioning structure does not allow a population to recover once it stabilises on high boldness and low vengefulness. In such situations, agents with average vengefulness score lower than others because of the high enforcement costs, which might be addressed by introducing a larger gap between enforcement and punishment costs. While Galan and Izquierdo [42] experiment with a set of reduced values for *meta* punishment and *meta* enforcement (but preserving the ratio), leading to norm collapse much earlier, their analysis is not extensive. Indeed, by preserving the ratio of costs but at lower values, it seems obvious that norms will collapse, because metanorms remain ineffective, as agents may pay a much higher price for enforcement than the punishment value they receive for not punishing.

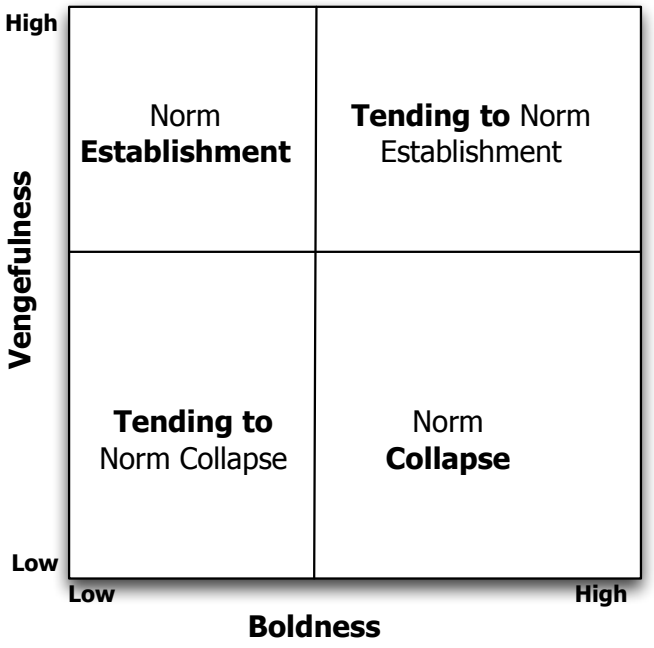


FIGURE 3.11: Characterising the vengefulness-boldness space

Through the deep analysis that has been performed in this chapter, various improvements are clearly needed in order to generalise the metanorm approach, and make it applicable over computational systems. First, the model assumes full control over the entire agent population, which allow accessing all agent scores and manipulating the population through the addition and elimination of agents. This, however, is rarely possible in computational systems, where populations are usually vast, and no single party can have control over every agent in the population. Furthermore, agent scores are usually private information that agents might choose not to share with others, which poses a further obstacle for the application of the approach suggested by Axelrod.

Second, the model assumes a fully connected network by which every agent is able to observe and interact with every other agent in the population. This is again an infeasible assumption for computational systems, since the network of such systems is usually governed by certain physical or logical constraints that allow to cope with the

heavy load and complexity generated from the vast amount of the comprising agents, which are unbearable if the agents are fully connected.

Finally, the model assumes a static punishment scheme by which each violation occurrence is responded to with exactly the same amount of punishment every time. This may work for some deterministic systems in which action effects are always the same. Thus, the gain from defection could also be predicted, and responded to with an appropriate amount of punishment. However, such static approach may not be suitable for dynamic complex systems, in which action effects could change over time and location. Moreover, the possibility of punishing the same violation by multiple agents adds another level of complexity to determining the right amount of punishment. If every agent responds with the same exaggerated punishment predetermined at the design time, then the violator will be over punished (i.e. the amount of punishment is much more than is actually required). As a result, this may discourage agents from participating in the system rather than encouraging them to change their behaviour.

These previous three points are the focus of the rest of the thesis and will be discussed in further details in the following chapters.

Chapter 4

Overcoming Omniscience in Axelrod's Metanorm Model

4.1 Introduction

Although Axelrod's investigation is successful in establishing cooperative norms, the model makes several assumptions that are unrealistic in real-world settings. In particular, in many domains it is not possible to remove unsuccessful agents and replicate those that are more successful, and there is no centralised control that could oversee this process. Instead, we need a mechanism through which individuals can learn to improve their strategies over time. If we enable individuals to compare themselves to others, and adopt more successful strategies, then we can take a *learning interpretation* of the evolutionary mechanism [106], without needing to remove and replicate individuals. However, this learning interpretation requires that the private strategies of individuals are available for observation by other agents, which is again an unreasonable assumption. As we have shown in the previous chapter, another issue with Axelrod's model

is inability to sustain cooperation over a large number of generations. Furthermore, Axelrod's approach relies on agents being able to punish both those that defect and those that fail to punish defection, yet this is unrealistic since it assumes *omniscience* through agents being aware of all norm violations and punishments.

In response, this chapter investigates alternatives that allow us to make use of the mechanisms resulting from Axelrod's investigations, in more realistic settings. Specifically, we first take a learning interpretation of evolution and describe an alternative technique, strategy copying, which prevents norm collapse in the long term. Second, we remove the assumption of omniscience and constrain the ability of agents to punish according to the defections they have observed. Finally, to obviate the need for knowledge of the private strategies of others, we propose a learning algorithm through which individuals improve their strategies based on their experience.

The rest of the chapter is organised as follows. In Section 4.2, we present our strategy copying technique, and show how it performs in the original context and in situations in which observation of defection is not guaranteed. In Section 4.3, we describe a reinforcement learning algorithm designed to avoid the need for access to the private strategies of others, before presenting our conclusions in Section 4.4, with a discussion of the significance of our results.

4.2 Strategy Copying

As indicated in Chapter 3, the evolutionary approach causes some problems in extended runs, leading to norm collapse. In addition, for use in domains such as peer-to-peer or wireless sensor networks, the agents themselves cannot be deleted or replicated, but instead must modify their own behaviour. In this section, therefore, we examine a simple alternative to Axelrod's model in which an agent that performs poorly in comparison

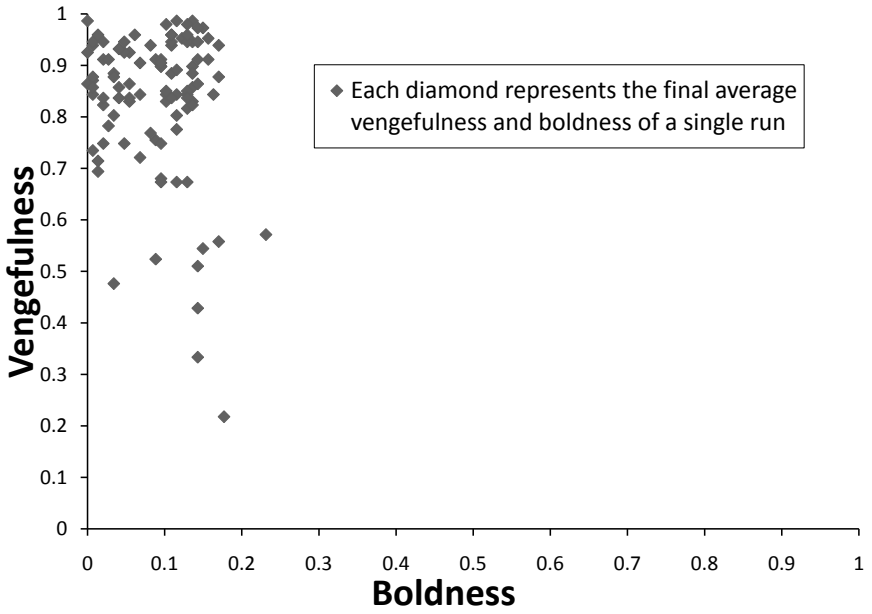


FIGURE 4.1: Strategy copying from the best agent, 100 timesteps

to others in the population can *learn* new strategies (in terms of vengefulness and boldness attributes) by adopting the strategy of other, better performing agents, replacing the existing strategy with a new one. Agents can achieve this in different ways: they can copy the strategy of the agent with the highest score or they can copy the strategy of one in a group of agents performing best in the population. It is important to note that the parameter set-up used in all experiments conducted in this section is the same as that specified in Table 3.2.

4.2.1 Strategy Copying from a Single Agent

Intuitively, copying the strategy of the agent with the highest score appears to be a promising approach. However, it leads to poor results in the long term because it draws strategies from only one agent rather than a population of agents. This makes the approach vulnerable to strategies that are only successful in a small number of

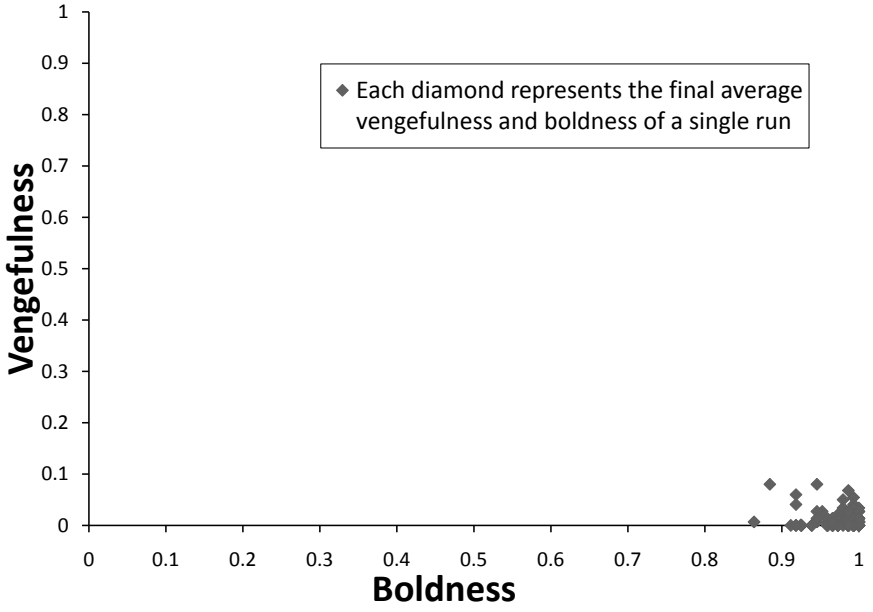


FIGURE 4.2: Strategy copying from the best agent, 1,000,000 timesteps

possible settings. Moreover, by failing to draw strategies from a variety of agents, the strategies tend to converge prematurely.

To illustrate, consider a group of students taking an examination, with one of the students having cheated. If the cheating student has not been seen, they may achieve the best exam performance. However, if all other students copy this behaviour and cheat in the next exam, there is a high possibility that they will be caught, and will thus suffer from much worse results than if they had not cheated. This is supported by the results shown in Figures 4.1 and 4.2, illustrating experiments with runs of 100 and 1,000,000 timesteps (where a timestep represents one *round* of agents having opportunities to defect and learning from the results, and is equivalent to a *generation* in the evolutionary approach). Each point on the graph (shown as a diamond to increase visibility) represents the average boldness and vengefulness of the population at the end of a single simulation run.

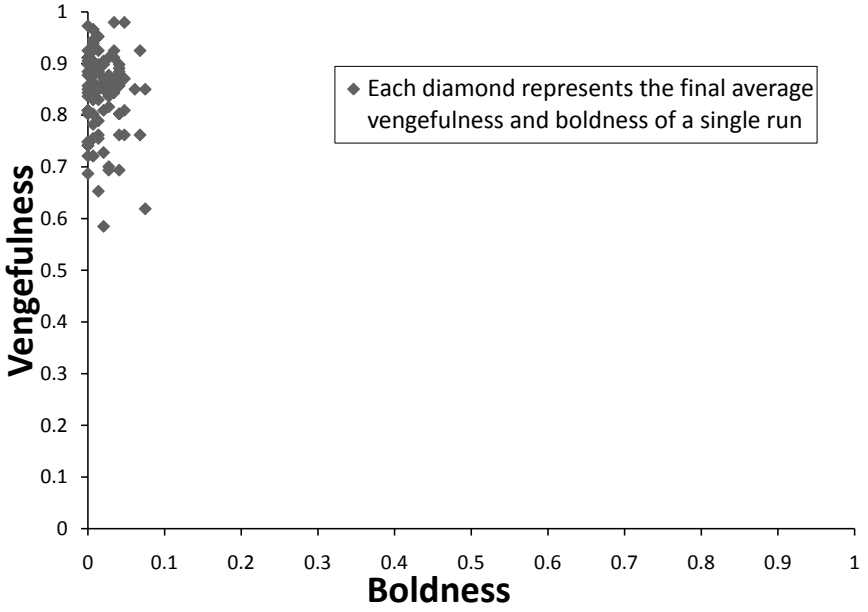


FIGURE 4.3: Strategy copying from a group of agents, 1,000,000 timesteps

In the short term, as can be seen from Figure 4.1, copying from the best agent leads to norm establishment. However, in the long term the norm collapses, as shown in Figure 4.2. This can be explained by the fact that an agent with low vengefulness that does not punish a defector (and thus does not pay an enforcement cost) but is also not metapunished, scores better than any other agent with high vengefulness that does punish (and thus pays the enforcement cost). As a result, other agents copy the low vengefulness of this agent so that low vengefulness becomes prevalent in the population. In the same way, when low vengefulness prevails in the population, an agent with high boldness defects, gaining a *temptation payoff*, and hurting others without receiving punishment. As a result, other agents copy the high boldness of this agent so that low vengefulness and high boldness is propagated through the population, leading to norm collapse.

4.2.2 Strategy Copying from a Group of Agents

Alternatively, as we have suggested, we might seek to copy the strategy of one in a group of high-performing agents. In this view, agents choose one agent, at random, from the group of agents with scores above the average, and copy its strategy. As previously, experiments of different durations (between 100 and 1,000,000 timesteps) were carried out; the results in Figure 4.3, for 1,000,000 timesteps, show that all runs ended with norm establishment in the long term, indicating that this approach is effective in eliminating the problematic effect of the replication method. This approach avoids norm collapse since it does not limit itself to the best performing agent, and thus does not run the risk of only adopting a strategy that performs well in a small number of settings.

4.2.3 Observation of Defection

In Axelrod's model (see Section 3.2), an agent z is able to punish another agent y that does not punish a defector x, even though agent z does not see the defection of agent x. However, such metapunishment is not possible if the original defection is not observed: guaranteed observation of the original defection is an unreasonable expectation in real-world settings. In consequence, our model needs adjustment so that metapunishment is only permitted if an agent observes the original defection. However, because this observation constraint limits the circumstances in which metapunishment is possible, its introduction corresponds to removing the metapunishment component from part of the game. In Axelrod's original experiments, metapunishment was introduced as a means to stabilise an established norm. In his setting, norms tend to collapse shortly after they are established without metapunishment. In fact, this remains the case in our model and our results confirm this.

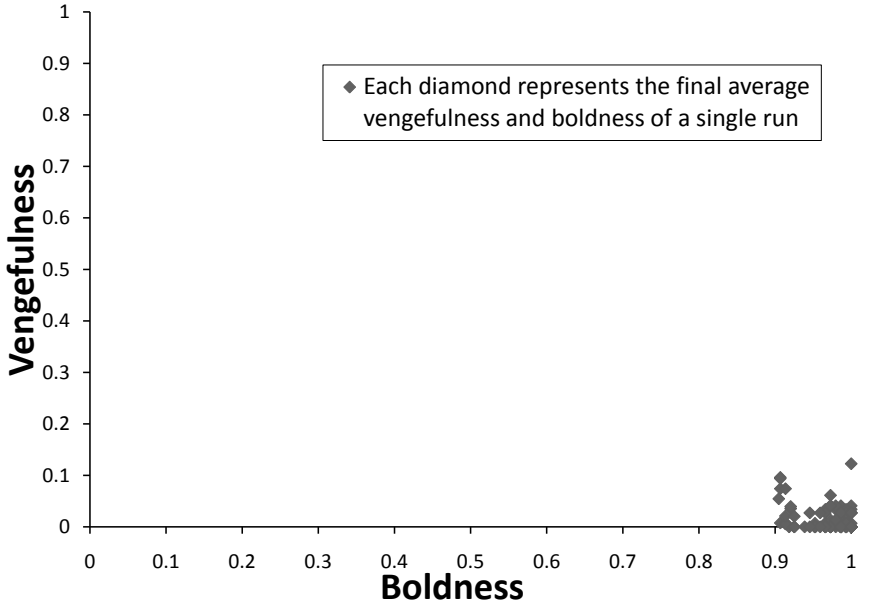


FIGURE 4.4: Strategy copying with defection observation constraint, 1,000,000 timesteps

More precisely, the observation constraint causes all runs to end in norm collapse when simulations are run for 1,000,000 timesteps, as shown in Figure 4.4. This is due to the fact that, as in the original model, runs initially stabilise on high vengefulness and low boldness, and then mutation causes vengefulness to reduce. If an agent y with high vengefulness and low boldness changes through mutation to give lower vengefulness, while boldness for all remains low, there is no defection and the mutated agent survives. If boldness then mutates to become just a little higher for a different agent x, with average vengefulness remaining high, x will still rarely defect because of relatively low boldness.

If it *does* defect, however, and *is* seen by others, it receives a low score, unless it is *not* punished, in which case the non-punishing agents may themselves be punished because of the high vengefulness in the general population. Here, agent y may not punish x because of the low probability of being seen (which must be below the low

boldness level to have caused a defection) or because it has mutated to have lower vengefulness. In the former case, y will not be metapunished for non-punishment, but in the latter case, y might be metapunished if it is seen by others. The likelihood of agent y's non-punishment being seen requires first x's defection being seen by y, and then y's non-punishment being seen by others. Importantly, in this new model, agents that metapunish y must themselves see x's defection. Since this combination of requirements is rare, such mutants like y survive for a longer duration, enabling their strategy to propagate through the population, and causing vengefulness to decrease. In addition, if another such event occurs, it will cause the average vengefulness of the population to drop further until it reaches a very low level. When the model runs over an extended period, such a sequence of events is much more likely, and low vengefulness allows a mutant of higher boldness to survive and spread among the whole population, which is the cause of the norm collapse.

4.3 Strategy Improvement

Once the observation constraint is introduced, strategy copying becomes inadequate. Furthermore, it requires that agents have access to the strategies and decision outcomes of others in order to enable the copying mechanism. As we have argued, in real-world settings such observations tend to be unrealistic. *Reinforcement learning* offers an alternative to Axelrod's evolutionary approach to improving performance of the society while keeping agent strategies and decision outcomes private. There are many reinforcement techniques in the literature, such as Q-learning [145], Policy Hill Climbing (PHC) [19] and WOLF-PHC [19], which we use as inspiration in developing a learning algorithm for strategy improvement in the metanorms game.

4.3.1 Q-learning

Q-learning is a reinforcement learning technique that allows the learner to use the (positive or negative) reward, gained from taking a certain action in a certain state, in deciding which action to take in the future in the same state. Here, the learner keeps track of a table of Q-values that records an action's quality in a particular state, and updates the corresponding Q-value for that state after each action. The new value is a function of the old Q-value, the reward received, and a learning rate, δ, and the action with the highest updated Q-value for the current state is chosen. However, for us, Q-learning suffers from two drawbacks. First, it considers an agent's past decisions and corresponding rewards, which are not relevant here; doing so would inhibit an agent's ability to adapt to new circumstances. Second, actions are precisely determined by the Q-value; there is no probability of action, unlike Axelrod's model.

Bowling and Veloso [19] proposed policy hill climbing (PHC), an extension of Q-learning that addresses this latter limitation. In PHC, each action has a probability of execution in a certain state, determining whether to take the action. Here, the probability of the action with the highest Q-value is increased according to a learning rate δ, while the probabilities of all other actions are decreased in a way that maintains the probability distribution, with each probability update occurring immediately after the action. In enhancing the algorithm, a *variable* learning rate is introduced, which changes according to whether the learner is winning or losing, inspired by the WOLF technique (win or learn fast). This suggests two possible values for δ: a low one to be used while an agent is performing well and a high one to be used while the agent is performing poorly.

However, in one round of Axelrod's game, an agent can perform multiple punishments (potentially one per defection and non-punishment observed), while only having a small number of opportunities to defect (four in Axelrod's configuration). Therefore, punishment and metapunishment actions would be considered much more frequently than

defection, leading to disproportionate update of probabilities of actions, with some converging more quickly than others. To address this imbalance, we can restrict learning updates to occur only at the end of each round, rather than after each individual action, so that boldness and vengefulness are reconsidered once in each round and evolve at the same speed. The aim here is to change the probability of action significantly when losing, while changing it much less when winning, providing more opportunities to adapt to good performance.

To sum up, while basic Q-learning is not appropriate because of the lack of action probabilities, WOLF-PHC suffers from a disproportionate update of such probabilities. Nevertheless, the use of the variable learning rate approach in PHC-WOLF is valuable in providing a means of updating the boldness and vengefulness values in determining which action to take. However, since agents that perform well need not change strategy, we can consider only one learning rate. The next section details our algorithm, inspired by this approach.

4.3.2 BV Learning

To address the concerns raised above, in this section, we introduce our BV learning algorithm. This requires an understanding of the relevant agent actions and their effect on the agent score, as summarised in Table 4.1, which outlines the different actions available to an agent and the consequence of each on the agent's score.

Now, since boldness is responsible for defection, an agent that obtains a good score as a result of defecting should increase its boldness, and an agent that finds defection detrimental to its performance should decrease its boldness. Learning suitable values for vengefulness is more complicated, since while it is responsible for both punishment and metapunishment, these also cause enforcement costs that decrease an agent's score.

TABLE 4.1: Effects of actions on agent score

Decision	**Effects**
Defect	Gain temptation payoff Hurt all other agents Potentially suffer punishment cost
Cooperate	—
Punish	pay enforcement cost
Not punish	Potentially suffer metapunishment cost
Metapunish	pay enforcement cost
Not metapunish	—

Low vengefulness allows an agent to avoid paying an enforcement cost, but can result in receiving metapunishment. Vengefulness thus requires a consideration of all these aspects.

First, in order to determine the unique effect of each individual action on agent performance, note that we decompose the single combined total score (TS) of the original model into distinct components, each reflecting the effect of different classes of actions. Therefore, each agent keeps track of four different utility values: the *defection score* (DS) incurred by an agent who defects, the *punishment score* (PS) incurred by an agent who punishes or metapunishes another (as a result of an enforcement cost), and the *punishment omission score* (POS) incurred by an agent who does not punish another when it should, and is consequently metapunished. In addition, these are combined into a total score (TS).

In this context, we can consider the algorithms used in our simulation, in two phases, as represented in Algorithms 2 and 3, called by Algorithm 1. (Note that we use subscripts to indicate the relevant agent only when needed.) More precisely, in Algorithm 2, each agent has various defection opportunities (o), and defects if its boldness is greater than the probability of its defection being seen (S_o). If an agent defects (Line 3), its DS increases by a *temptation payoff*, T (Line 4), but it *hurts* all others in the population,

Algorithm 1 The Simulation Control Loop: *simulation*$(T, H, P, E, \gamma, \delta)$

```
1. for each round do
2.     interact(T, H, P, E)
3.     learn(γ, δ)
```

Algorithm 2 interact(T, H, P, E)

```
1.  for each agent i do
2.      for each opportunity to defect o do
3.          if B_i > S_o then
4.              DS_i = DS_i + T
5.              for each agent j : j ≠ i do
6.                  TS_j = TS_j + H
7.                  if see(j,i,S_o) then
8.                      if punish (j, i, V_j) then
9.                          DS_i = DS_i + P
10.                         PS_j = PS_j + E
11.                     else
12.                         for each agent k : k ≠ i ∧ k ≠ j do
13.                             if see(k,j,S_o) then
14.                                 if punish (k, j, V_k) then
15.                                     PS_k = PS_k + E
16.                                     POS_j = POS_j + P
```

whose scores decrease by H (line 6), where H is a negative number that represents the hurt suffered. If an agent cooperates, no scores change. DS thus determines whether an agent should increase or decrease boldness in relation to its utility.

Each hurt agent can in turn observe the defection and react to it with punishment that is probabilistic to its vengefulness. Punishment and metapunishment both have two-sided consequences: if an agent j sees agent i defect in one of its opportunities (o) to do so, with probability S_o (Line 7), and decides to punish it (which it does with probability V_j; Line 8), i incurs a punishment cost, P, to its DS (Line 9), while the punishing agent incurs an enforcement cost, E, to its PS (Line 10). If j does not punish i, and another agent k sees this in the same way as previously (Line 13), and decides to metapunish (Line 14), then k incurs an enforcement cost, E, to its PS, and j incurs a punishment cost P to its POS. Note that both P and E are negative values, so they are added to the total to reflect their effect.

In Axelrod's original model, those agents that are one standard deviation or more below the mean are eliminated and replaced in the subsequent population generation with new agents following the strategy captured by the boldness and vengefulness values of those agents that are one standard deviation or more above the mean. Thus, poorly performing agents are replaced by those that perform much better. In contrast, in our model, in Algorithm 3, we distinguish more simply between good and poor performance, with only agents that score below the mean reconsidering their strategy. Thus, for each agent, we combine the various component scores into a total, TS and, if the agent is performing poorly (in relation to the average score, $AvgS$ in Line 7), we reconsider its boldness and vengefulness. Note that this average score is established through lines 1-5 in the algorithm.

Now, in order to ensure we allow a degree of exploration (similar to mutation in the original model's evolutionary approach, to provide comparability) and to enable an agent to step out of the learning trend, here we adopt an *exploration rate*, γ, which regulates adoption of random strategies from the available strategies universe (Line 8). If an agent does not explore then, if defection is the cause of a low score (Line 12), the agent decreases its boldness, and increases it otherwise. Similarly, agents decrease their vengefulness if they find that the effect of punishing is worse than the effect of not punishing (Line 22), and increase vengefulness if the situation is reversed. As both PS and POS represent the result of two mutually exclusive actions, their difference for a particular agent determines the change to be applied to vengefulness. For example, if $PS > POS$, then punishment has some value, and vengefulness should be increased.

Finally, given a decision on whether to modify an agent's strategy, the degree of the change, or *learning step* (δ), must also be considered. Since vengefulness and boldness have eight possible values from $\frac{0}{7}$ to $\frac{7}{7}$, we adopt the conservative approach of increasing or decreasing by one level at each point, corresponding to a learning rate of $\delta = \frac{1}{7}$. Thus,

Algorithm 3 learn(γ, δ)

```
1.  Temp = 0
2.  for each agent i do
3.     TS_i = TS_i + DS_i + PS_i + POS_i
4.     Temp = Temp + TS_i
5.  AvgS = Temp/no_agents
6.  for each agent i do
7.     if TS_i < AvgS then
8.        if explore(γ) then
9.           B_i = random()
10.          V_i = random()
11.       else
12.          if DS_i < 0 then
13.             if B_i − δ < 0 then
14.                B_i = 0
15.             else
16.                B_i = B_i − δ
17.          else
18.             if B_i + δ > 1 then
19.                B_i = 1
20.             else
21.                B_i = B_i + δ
22.          if PS_i < POS_i then
23.             if V_i − δ < 0 then
24.                V_i = 0
25.             else
26.                V_i = V_i − δ
27.          else
28.             if V_i + δ > 1 then
29.                V_i = 1
30.             else
31.                V_i = V_i + δ
```

an agent with boldness of $\frac{5}{7}$ and vengefulness of $\frac{3}{7}$ that decides to defect less and punish more will decrease its boldness to $\frac{4}{7}$ and increase its vengefulness to $\frac{4}{7}$.

4.3.3 Evaluation

The algorithm is designed to mimic the behaviour of Axelrod's evolutionary approach as much as possible, while relaxing Axelrod's unrealistic assumptions. This allows us to replicate Axelrod's results and investigate his approach in more realistic problem domains, as shown in Figure 4.5. The parameter set-up used in all experiments carried

out in this section is shown in Table 4.2, with the addition of parameters related to the *BV Learning* algorithm.

TABLE 4.2: Parameter initialisation

Term	**Description**	**Value**
i, j	Individuals	A number to identify individual agents
S	Probability of a defection being seen by any given individual	Uniform distribution from 0 to 1
B_i	Boldness of i	Uniform distribution from 0 to 1
V_i	Vengefulness of i	Uniform distribution from 0 to 1
T	Player's temptation to defect	$+3$
H	Hurt suffered by others as a result of an agent's defection	-1
P	Cost of being punished	-9
E	Enforcement cost, i.e. cost of applying punishment	-2
P'	Cost of being punished for not punishing a defection	-9
E'	Cost of punishing someone for not punishing a defection	-2
δ	*learning step*	$\frac{1}{7}$
γ	*exploration rate*	0.01

The analysis of a sample run reveals that agents with low vengefulness and agents with high boldness start changing their strategies. Here, agents with high boldness defect frequently, and are punished as a result, leading to a very low *DS*, in turn causing these agents to decrease their boldness. Agents with low vengefulness do not punish and are consequently frequently metapunished; as a result, their *PS* is much better (lower in magnitude) than their *POS*, causing them to increase their vengefulness. The population eventually converges to comprise only agents with high vengefulness and low boldness. While noise is still introduced via the exploration rate causing random strategy adoption, the learning capability enables agents with such random strategies to adapt quickly to the trend of the population.

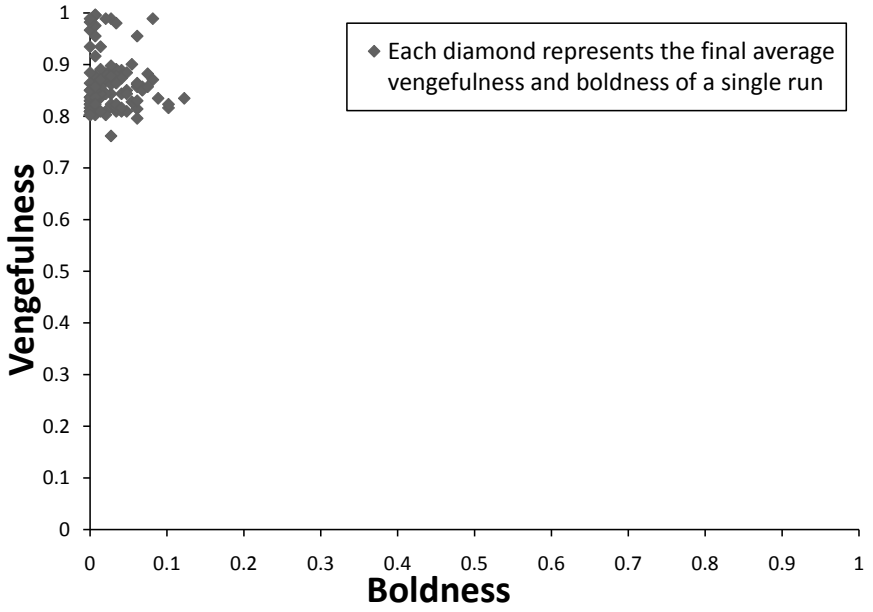


FIGURE 4.5: Strategy improvement (with $\gamma = 0.01$), 1,000,000 timesteps

As before, we also consider the problem of ensuring that an original defection is observed in order to provide a metapunishment. Introducing this constraint into our new algorithm, we ran experiments over different periods, with results indicating that norm establishment is robust in all runs. An example run for 1,000,000 timesteps is shown in Figure 4.6. This is because agents that use this new learning algorithm only change their strategy incrementally without wholesale change at any single point. The effect of a mutant with low vengefulness is not significant since, while the mutant might survive for a short period and cause some agents to change their vengefulness, any such change will be slight. It thus does not prevent such agents from detecting the mutant subsequently, in turn causing the mutant to increase its vengefulness.

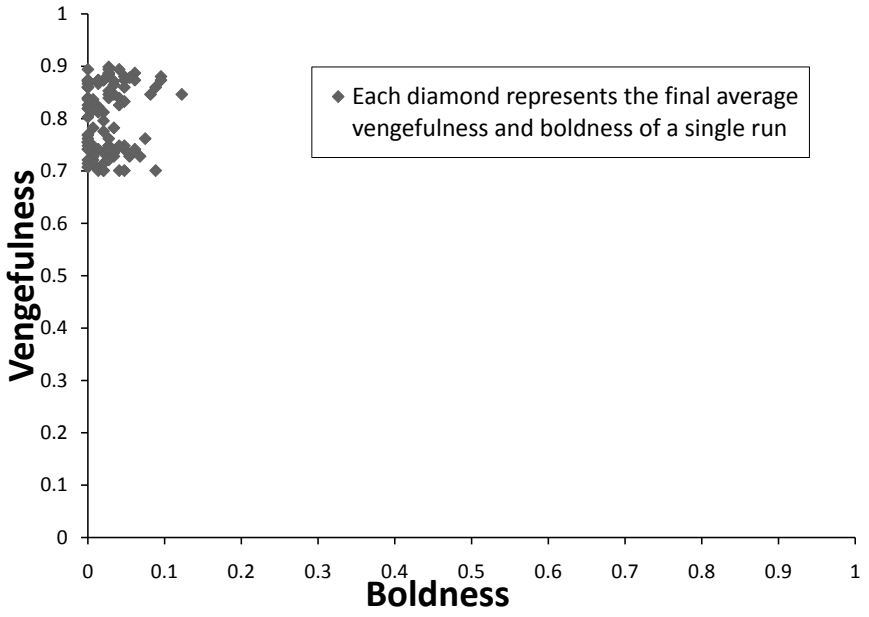


FIGURE 4.6: Strategy improvement with defection observation constraint (with $\gamma = 0.01$), 1,000,000 timesteps

4.4 Conclusion

In systems of self-interested autonomous agents we often need to establish cooperative norms to ensure the desired functionality. Axelrod's work on norm emergence gives valuable insight into the mechanisms and conditions in which such norms may be established. However, there are two major limitations. First, as we have shown in Chapter 3, and explained in detail, norms collapse even in the metanorms game for extended runs. Second, the model suffers from limitations relating to assumptions of omniscience. In response, this Chapter has (i) investigated those aspects of Axelrod's investigation that are unreasonable in real-world domains, and (ii) proposed *BV learning* as an alternative mechanism for norm establishment that avoids these limitations.

More specifically, we replaced the evolutionary approach with a learning interpretation

in which, rather than remove and replicate agents, we allow them to learn from others. Two techniques were considered: copying from a single agent and copying from a group. The former suffers the same problems of long term norm collapse associated with Axelrod's approach but, by avoiding strategies that only perform well in restricted settings, the latter addresses the problems and brings about norm establishment. In addition, we addressed Axelrod's assumption of omniscience, in which agents considering metapunishment are not explicitly required to *see* the original defection. By doing so, however, the metapunishment activity in the population, for stabilising an established norm, decreases and leads to norm collapse.

Since learning strategies from *others* (either individuals or groups) is unable to establish norms for cooperation (and is, in addition, unrealistic since it assumes that agent strategies are not private), we have developed an alternative, *BV learning*, in which agents learn from their *own* experiences.

Through this approach we have shown that not only it is possible to avoid the unrealistic assumption of knowledge of others' strategies, but also that by enabling individuals to incrementally change their strategies we can avoid norm collapse, even with observation constraints on metapunishment.

Next, our aim is to focus on applying the model to interaction networks in order to analyse how different network structures can impact on the achievement of norm emergence. In particular, our current model is limited in that the algorithm relies on agents comparing their own score to the average score of all other agents to determine if learning is warranted. This constrains our move towards turning Axelrod's model into something more suitable for real-world distributed systems and, in consequence, we aim to enable agents to estimate their learning needs based on their own, individual, experience by monitoring their past performance. Moreover, we also investigate the possibility of integrating *dynamic* punishments (in Chapter 6), rather than the current

static ones (that are fixed regardless of what has happened), by which agents can modify the punishments they impose on others according to available information about the severity of violation, or according to whether the violating agent is a repeat offender, and if so, how many times.

Chapter 5

Establishing Norms for Network Topologies

5.1 Introduction

Axelrod's model is interesting and valuable in examining how norms can be established in a population of agents. Using our simulation model of Chapter 4, we are able to match Axelrod's results (and in fact improve on them for extended runs, even with an observability constraint). Although this provides a valuable illustration of the value of metanorms in avoiding norm collapse in a system in which there is no central control, it still assumes a fully connected network (in which each agents is connected to every other agent), and omits, similarly to Axelrod's model, consideration of the important aspect of network topology, which is a main characteristic of real world domains [93]. In particular, in real-world domains, such as peer-to-peer and wireless sensor networks, the network of agents is not fully connected, with agents tending to interact with a

small subset of others on a regular basis, yet it is only through such interactions that defection can be observed and punishment administered.

Some work has already been undertaken on examining the impact of different topologies on norm establishment. For example, Savarimuthu et al. [112] consider the *ultimatum game* in the context of providing advice to agents on whether to change their norms in order to enhance performance for random and scale-free networks. Delgado et al. [33] study norm emergence in coordination games in scale-free networks, and Sen et al. [115] examine rings and scale-free networks in a related context. Additionally, Villatoro et al. [140] explore norm emergence with memory-based agents in lattices and scale-free networks.

While these efforts provide valuable and useful results, the context of application has been limited, with only two agents involved in each encounter, rather than a larger population of agents. This simplifies the problem when compared with those in which the actions of multiple interacting agents can impact on norm establishment. In response, this chapter builds on Axelrod's model enhancement that is presented in Chapter 4 to address the context of different topological configurations. First, the model so far assumes a fully connected network, and is predicated on that for certain aspects, such as how one agent observes another's actions. In a variably connected structure, this part of the model is thus not meaningful and requires modification, causing some difficulties in establishing norms. Second, in scale-free networks, which contain both heavily connected nodes (*hubs*) and lightly-connected nodes (*outliers*), hubs obstruct norm emergence since they require observation of, and interaction with, so many others in the network, causing asymmetric effects.

The rest of the chapter is organised as follows. Section 5.2 outlines the metanorms game, adjusted to suit the purposes of this chapter, and augmented with a learning mechanism. Sections 5.3 and 5.4 describe in detail the impact of applying the model

in lattices and small world, respectively. In Section 5.5, we consider the problems that arise from the use of scale-free networks, and presents an adaptation of the model that copes with their characteristics. Section 5.6 introduces our solution for achieving norm emergence in this context and, finally, Section 5.7 concludes this chapter.

5.2 Imposing Topologies on Metanorms

While Axelrod's model assumes a fully connected network, an unlikely and unreasonable assumption, other network topologies must instead be considered, reflecting different potential configurations of agents, in which agents are connected only to a subset of other agents, their *neighbours*. This constraint on connectivity between agents implies some adjustments to Axelrod's model, as follows.

First, in Axelrod's model it is assumed that an agent's defection penalises all other agents in the population. The introduction of a topology enables us to restrict the penalty to only those agents with which the defector interacts. Second, in Axelrod's model, agents are assumed to be able to observe the entire population. By introducing a topology, we employ a more realistic model in which an agent can only observe those agents with which it interacts. Third, punishment requires observation of misbehaviour. In Axelrod's model, this requirement is implicit as it makes no meaningful distinction. However, by introducing constraints on observation and rendering the model more realistic, a further refinement is required: an agent can only punish a defector if the agent can observe the defector. In addition, an agent can only metapunish an agent that fails to punish a defector if it can observe both the defector *and* the agent that fails to punish the defector. Finally, in order to enhance an agent's individual performance, it compares itself to others in the population before deciding whether to modify its

Algorithm 4 The Simulation Control Loop: *simulation*$(T, H, P, E, \gamma, \delta)$

```
1. for each round do
2.    interact(T, H, P, E)
3.    learn(γ, δ)
```

Algorithm 5 interact(T, H, P, E)

```
1. for each agent i do
2.    for each opportunity to defect o do
3.       if B_i > S_o then
4.          DS_i = DS_i + T
5.          for each agent j ∈ NB_i do
6.             TS_j = TS_j + H
7.             if see(j,i,S_o) then
8.                if punish (j, i, V_j) then
9.                   DS_i = DS_i + P
10.                  PS_j = PS_j + E
11.               else
12.                  for each agent k ∈ NB_j : k ≠ i do
13.                     if see(k,j,S_o) then
14.                        if punish (k, j, V_k) then
15.                           PS_k = PS_k + E
16.                           POS_j = POS_j + P
```

strategy. However, since agents can only observe their neighbours, these are the only agents they are able to learn from.

In consequence, the algorithms presented in Chapter 4 are no longer adequate, and need to be replaced with Algorithms 4, 5, and 6. Specifically, the changes are as follows. First, in Algorithm 5, Line 5 considers only agent i's neighbours NB_i rather than all the agents in the population, and Line 12 considers only agent j's neighbours NB_j. In Algorithm 6, the average score in Line 3, $AvsS_{NB_i}$ refers to the average score of agent i's neighbourhood NB_i (that is, those agents to which agent i is connected). In this way, and with these simple modifications, our algorithms now address the needs of different topological structures.

In what follows, we consider these modifications to the model in the context of different kinds of topologies, in particular small world models and scale-free networks. However, to start, we introduce lattices, since they provide the foundation on which small-worlds

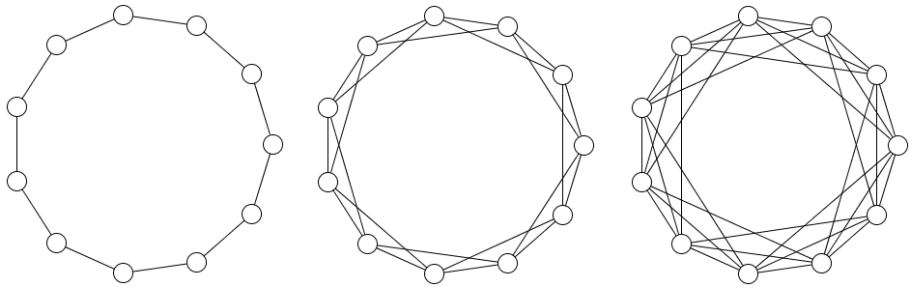
Algorithm 6 learn(γ, δ)

```
1.  for each agent i do
2.    TS_i = TS_i + DS_i + PS_i + POS_i
3.    if TS_i < AvgS_{NB_i} then
4.      if explore(γ) then
5.        B_i = random()
6.        V_i = random()
7.      else
8.        if DS_i < 0 then
9.          if B_i − δ < 0 then
10.           B_i = 0
11.         else
12.           B_i = B_i − δ
13.       else
14.         if B_i + δ > 1 then
15.           B_i = 1
16.         else
17.           B_i = B_i + δ
18.       if PS_i < POS_i then
19.         if V_i − δ < 0 then
20.           V_i = 0
21.         else
22.           V_i = V_i − δ
23.       else
24.         if V_i + δ > 1 then
25.           V_i = 1
26.         else
27.           V_i = V_i + δ
```

are based. Note that all experiments in this chapter have been performed with the same parameter set-up as specified in Table 4.2.

5.3 Metanorms in Lattices

A lattice (typically a simple ring structure) is perhaps the simplest network topology we consider, in particular, because it is also used as a base for more interesting and valuable topologies. In a (one-dimensional) lattice with neighbourhood size n, agents are situated on a ring, with each agent connected to its neighbours n or fewer hops (lattice spacings) away, so that each agent is connect to exactly $2n$ other agents. Thus,



(a) NB Size 1 (b) NB Size 2 (c) NB Size 3

FIGURE 5.1: Lattice Topologies

in a lattice topology with $n = 1$, each agent has two neighbours and the network forms a ring as shown in Figure 5.1(a). In a lattice topology with $n = 3$, each agent is connected to 6 neighbours, as shown in Figure 5.1(c).

5.3.1 Neighbourhood Size

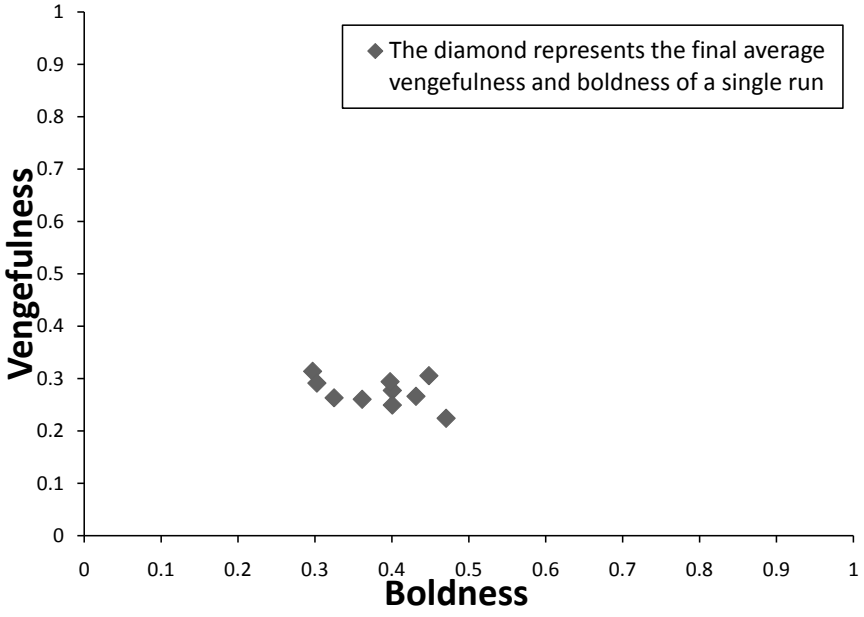
It is clear that, depending on the neighbourhood size, lattices may be more or less connected. Those with larger neighbourhood sizes are more similar to Axelrod's fully connected model; our hypothesis is that as the neighbourhood size increases, the greater connections between agents enable punishment and metapunishment to become more effective in reducing boldness and increasing vengefulness. In order to investigate this hypothesis, we ran several experiments.

In our first set of experiments, we used 51 agents (so we have an even number, plus one, to account for the $2n$ neighbours plus our original agent), and varied the neighbourhood size between the least connected lattice (the ring topology) and the most connected lattice ($n = 25$). Each experiment involved 10 separate runs, with each run comprising 1,000,000 timesteps for a particular neighbourhood size.

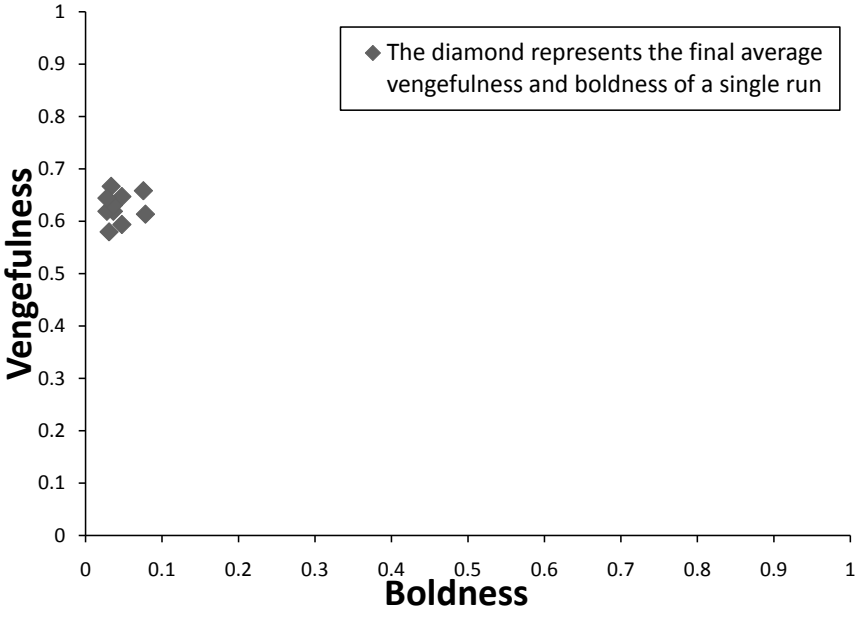
For the least connected lattice (n of 1), no norm is established, as runs ended in both relatively low boldness and relatively low vengefulness (see Figure 5.2(a)). In this case, though agents rarely defect, they also rarely punish a defection. This constitutes an unstable situation in which defecting could be a rewarding behaviour for agents as it is relatively unlikely to be penalised. However, increasing the neighbourhood size slightly to 3 (Figure 5.2(b)) has a noticeable impact on the results, as the boldness of the population drops almost to 0, which means that agents do not defect. While the level of vengefulness increases, it is still not at a level that can be considered to correspond to norm emergence, since agents might still not punish a defection without being metapunished for not doing so.

In addition, increasing the neighbourhood size to 13 has the same effect on boldness and a stronger effect on vengefulness (see Figure 5.3(a)), as vengefulness increases further, and almost to its maximum, of 1, when the neighbourhood size of 19 is used (see Figure 5.3(b)). These results suggest that increasing neighbourhood size strengthens norm emergence, by virtue of agents being more willing to punish norm violators.

In seeking to provide more detail for analysis, the results of all runs were averaged, and shown on the graph in Figure 5.4, with neighbourhood size plotted against boldness and vengefulness. This shows that a neighbourhood size as small as 2 is enough to maintain boldness near 0, indicating that agents do not defect except when they *explore* as a result of sometimes adopting random strategies (introduced for comparability with Axelrod's model). Conversely, increasing the neighbourhood size has a major impact on vengefulness, until the neighbourhood size reaches around 15 (at which point an agent is connected to half the population) when it brings only very minor change. This is because, in a poorly connected environment, agents that do not punish defection can more easily escape metapunishment than in a more connected environment.

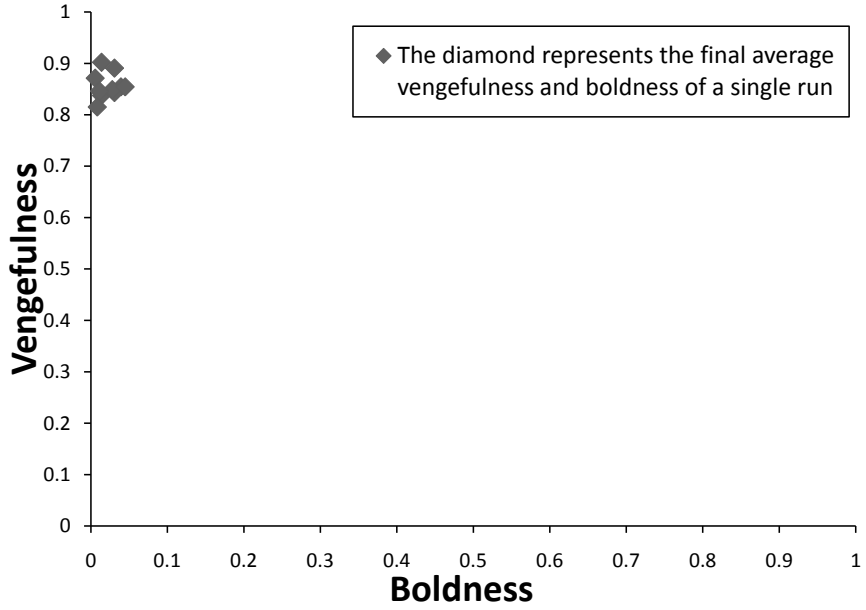


(a) Lattice with neighbourhood size 1, and 1,000,000 timesteps

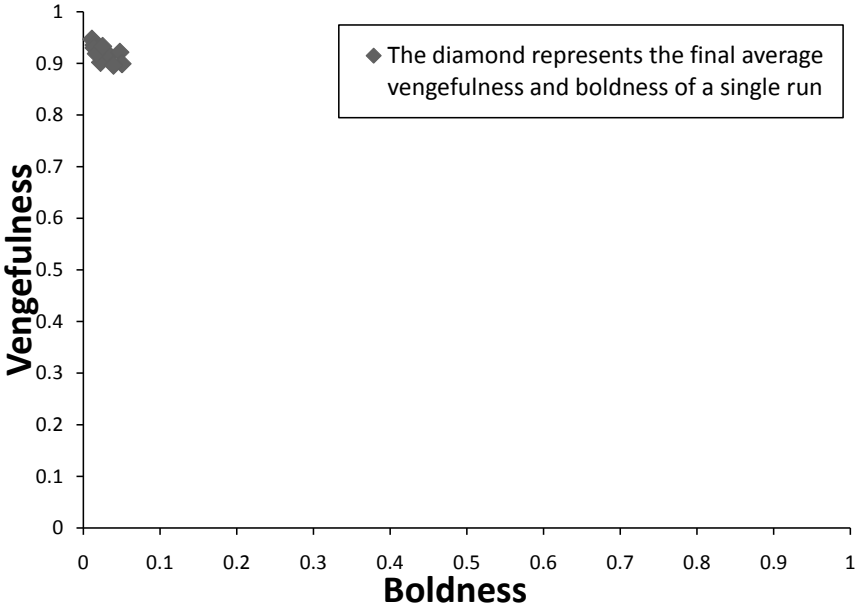


(b) Lattice with neighbourhood size 3, and 1,000,000 timesteps

FIGURE 5.2: Lattice Topologies



(a) Lattice with neighbourhood size 13, and 1,000,000 timesteps



(b) Lattice with neighbourhood size 19, and 1,000,000 timesteps

FIGURE 5.3: Lattice Topologies

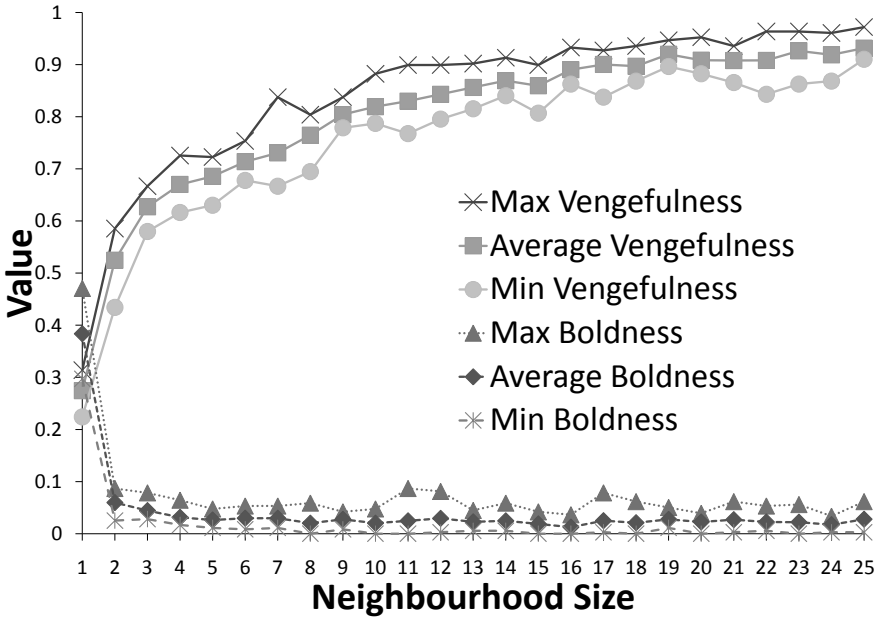


FIGURE 5.4: Lattice: impact of neighbourhood size on final B and V

As we hypothesised, increasing neighbourhood size brings a corresponding effect on the strategy of agents (in terms of boldness and vengefulness). Only the most poorly connected lattices have moderate levels of boldness, with vengefulness increasing monotonically over a longer period before it stabilises at a level consistent with norm establishment. The connections between agents give rise to this behaviour, with an increase in connections providing more opportunities for agents to respond to defectors appropriately.

5.3.2 Population Size

Now, if we increase the population size while keeping the neighbourhood size static, we decrease the relative number of connections among the overall population. This suggests that convergence to norm establishment should decrease, in line with the results obtained above. In the second set of experiments, therefore, the neighbourhood

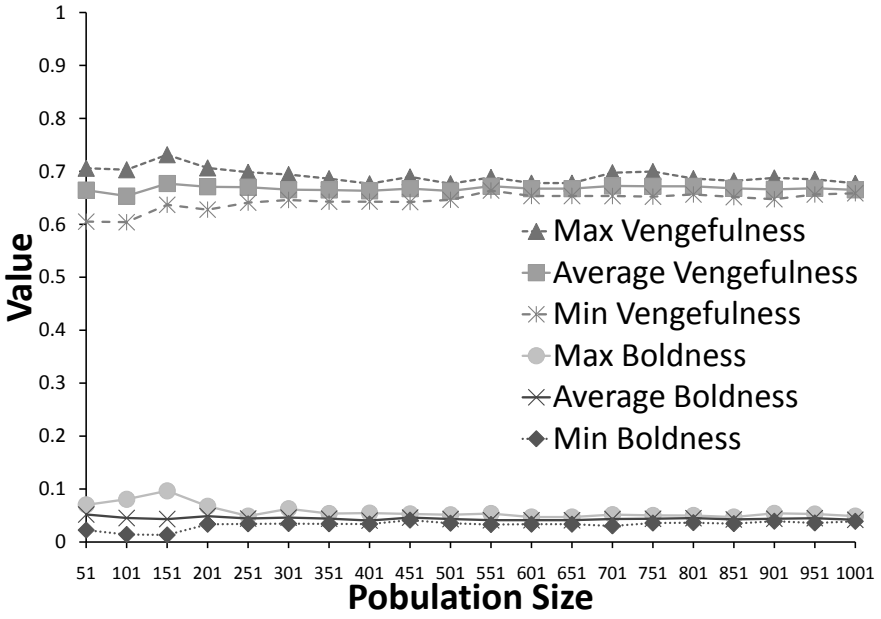


FIGURE 5.5: Lattice: impact of population size on final B and V (where neighbourhood size, $n = 3$)

size was fixed and the population size varied between 51 and 1,001 agents. However, the results obtained, shown in Figure 5.5 for a neighbourhood size of 3 (though other values gave similar results), are not as expected, and suggest that increasing the population size has no effect on the rate of norm emergence, as all runs for all sizes of population end almost with the same level of boldness and vengefulness.

These results suggest that norm emergence in a community of agents that interact in a lattice is not affected by total population size but by neighbourhood size. By increasing the number of neighbours, norm establishment becomes more likely, irrespective of the size of the population. In other words, the likelihood of norm establishment is governed by the total amount of punishment that could potentially be brought upon a defector or an agent failing to punish a defector, which may be termed the *potential peer pressure* of a lattice. This is because such lattices essentially comprise multiple overlapping localities in which agents are highly connected: via punishments, the agents in these

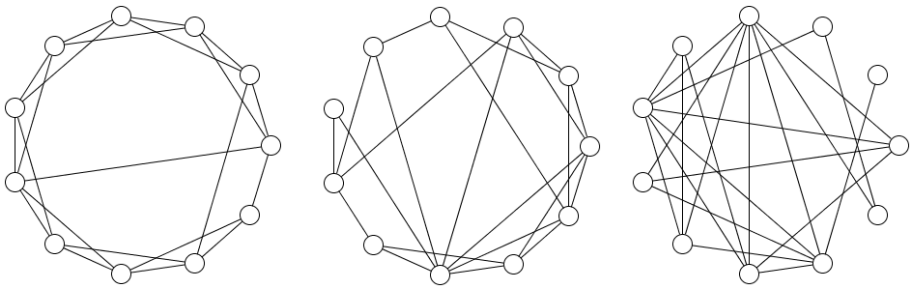
localities impose a strong influence on their neighbours. Increasing the population size simply increases the number of such overlapping regions.

5.4 Metanorms in Small World

While lattices are regular structures, as opposed to random structures, Watts and Strogatz [146] noted that many biological, technological and social networks lie somewhere between the two: neither completely regular nor completely random. They instead proposed *small world networks* as a variation of lattices in which agents are connected to others n or fewer hops (on the ring) away, but with some of the connections replaced by connections to other randomly selected nodes in the network, in line with some specific rewiring probability (RP). Examples of such networks with different rewiring probabilities are shown in Figure 5.6.

Thus, while lattices essentially create overlapping localities of well connected agents (since agents are connected to $2n$ agents immediately surrounding them), the effect of small world is to break these connections. Though the number of connections does not change, the locality effect does, since there may no longer be localities of well connected agents, but instead agents with some connections to their local neighbours, and some connections to others elsewhere in the network. As these local regions break down, the strong influence of an agent's local neighbours, causing compliance with norms, should also break down because of the more sparse connections.

To verify this hypothesis, we investigated the impact of the rewiring probability by running the model with different values, in populations of 51 agents, for different neighbourhood sizes. The results of the experiment with a neighbourhood size of 3 are shown in Figure 5.7, which indicates that increasing the RP decreases the final



(a) RP = 0.1 (b) RP = 0.2 (c) RP = 0.8

FIGURE 5.6: Small World Topologies

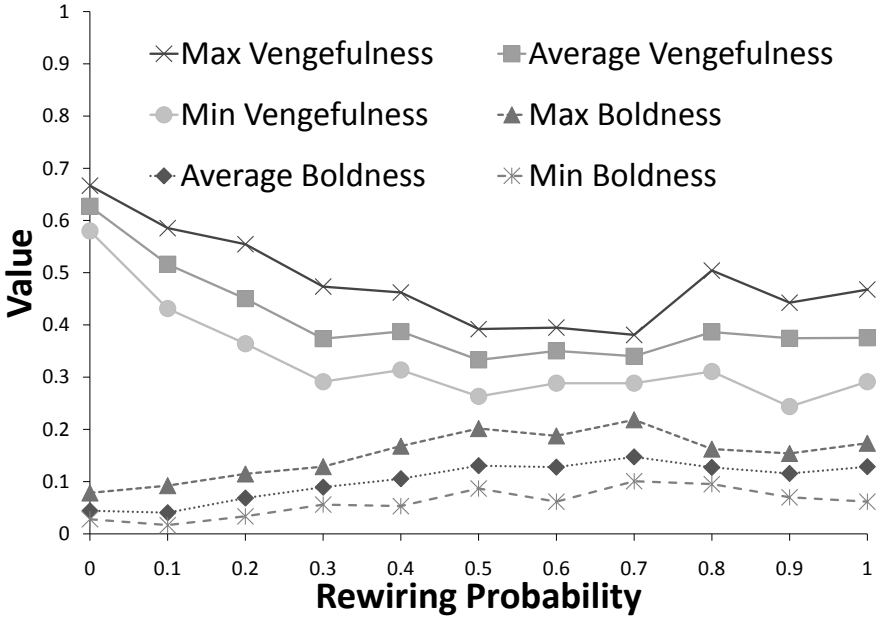


FIGURE 5.7: Small world: impact of rewiring on final B and V (where neighbourhood size, $n = 3$)

average vengefulness in the population, with other neighbourhood sizes giving similar results.

The results obtained are due to the fact that, as a result of rewiring, agents no longer affect just their locality, but now affect agents that are much further away, consequently requiring establishment of the norm in multiple localities. For example, in the case of

neighbourhood size of 3, it is clear that not only is the norm not established, but as the RP rises above small values, the trend moves further away from establishment, since the connections of agents are increasingly rewired, giving a locality effect similar to lattices with a neighbourhood size of 2 (discussed in Section 5.3.1). In addition, rewiring to other agents further away brings the need to establish the norm in all those localities to which an agent is connected, making it much more difficult.

In terms of boldness, it is clear that the RP of small world does not impact on the level of defection in the population since, independently, boldness remains very low, indicating that agents are very unlikely to defect.

5.4.1 Neighbourhood Size and Rewiring Probabilities

As discussed in Section 5.3.1, increasing neighbourhood size causes an increase in vengefulness in lattices. In seeking to understand the impact in a small world, we repeated the lattice experiments in this new context, for different values of the RP. Results for a rewiring probability of 0.4 are shown in Figure 5.8 (with results for other values of the RP being similar in trend), again showing that neighbourhood size increases vengefulness. However, note that, in comparison to lattices, vengefulness in a small world is lower for the same neighbourhood size. This is because the agents must now respond to defections in different regions of the network, where there is less influence on behaviour, and thus potentially incurring greater enforcement costs.

5.4.2 Population Size and Rewiring Probabilities

Population size has been shown to have no effect on norm establishment in lattices due to the *potential peer pressure* arising from the multiple overlapping localities. However, since these concentrated local regions of connected agents are weakened in small world,

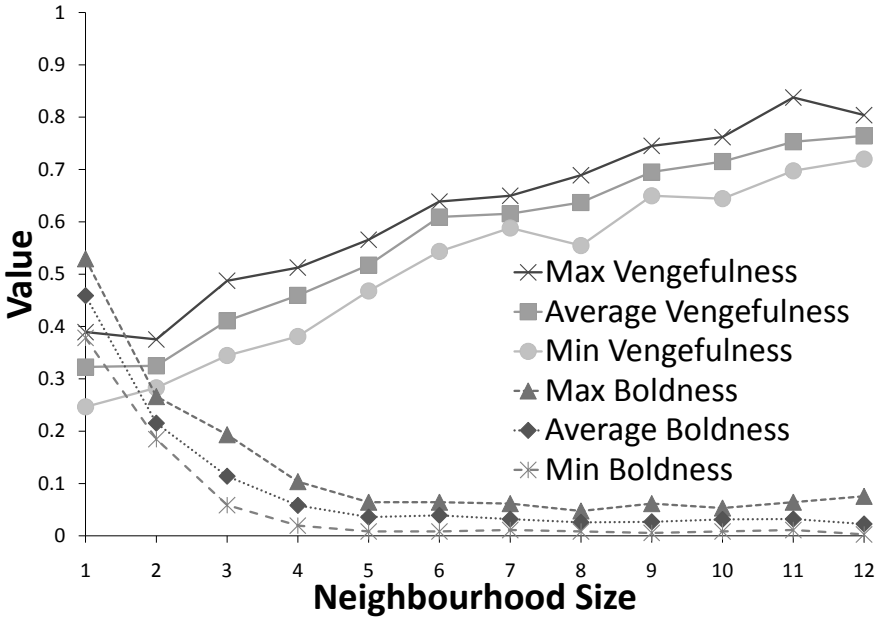


FIGURE 5.8: Small world: impact of neighbourhood size on final B and V (RP=0.4)

we repeated the previous experiments to determine the effect with RPs of 0.2, 0.4, 0.6, 0.8 and 1.0, and n of 5. The results indicate that boldness is not affected by the changes of the population size as boldness is always close to zero, as shown in Figure 5.9, but vengefulness decreases as the RP increases. More specifically, when the RP is 0.2, increasing the population size has little effect, as shown in Figure 5.10. However, for the other RP values, increasing the population size decreases vengefulness. Again, this is due to rewiring breaking down the strong locality effect, and this is magnified with increasing population sizes, since there is a greater opportunity for connections to other localities, causing a greater cost for agents seeking to bring about norm establishment in all these localities at once.

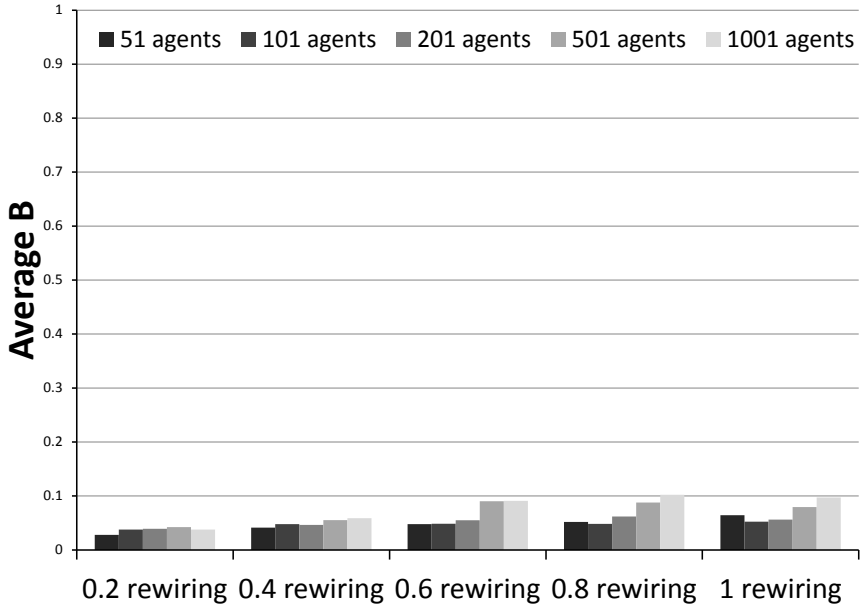


FIGURE 5.9: Small world: impact of rewiring and population size on final boldness (where neighbourhood size $n = 5$)

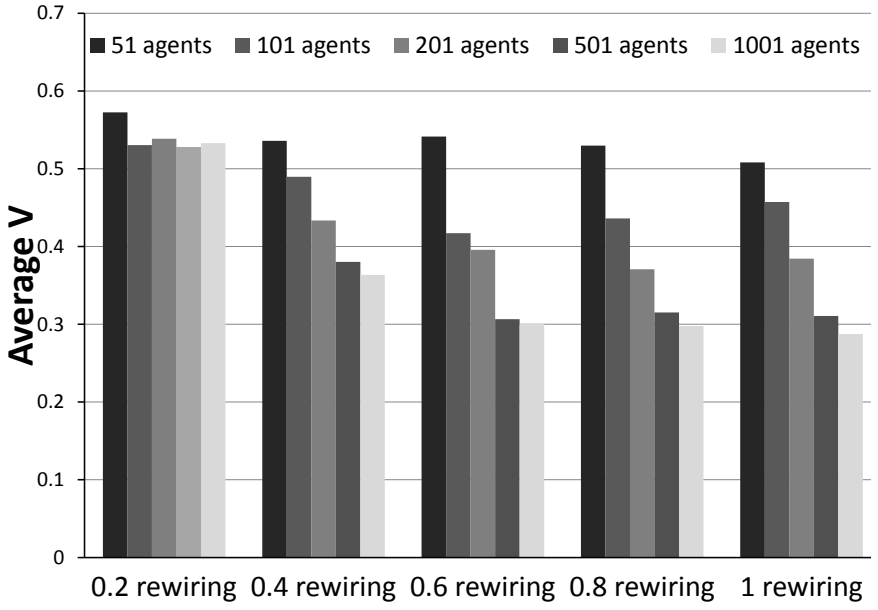


FIGURE 5.10: Small world: impact of rewiring and population size on final vengefulness (where neighbourhood size $n = 5$)

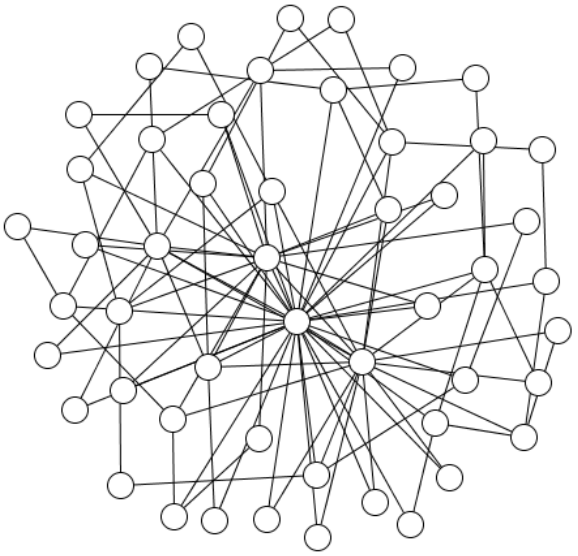


FIGURE 5.11: Scale-free network

5.5 Metanorms in Scale-free Networks

The topologies considered above are similar in that each agent has exactly the same number of connections, in contrast to scale-free networks [9], in which connections between nodes follow the power law distribution. Thus, some nodes have a vast number of connections, but the majority have very few connections, as illustrated in Figure 5.11. These properties of scale-free networks suggest an imbalance in connections. In turn, this has an impact on the results that can be obtained, due both to punishment and to enforcement costs, which dramatically modify the dynamics of the system. To investigate this, we ran 1000 experiments on a scale-free network with 1000 agents, five of which were *hubs* (having a large number of connections) and the others (which we call *outliers*) with at least two connections to other agents in the population, and typically no more than four connections (according to Barabasi's algorithm [9]). Each experiment was run for 1,000,0000 timesteps, and parameters for the experiments were as follows (and are the same for all subsequent experiments reported in this chapter):

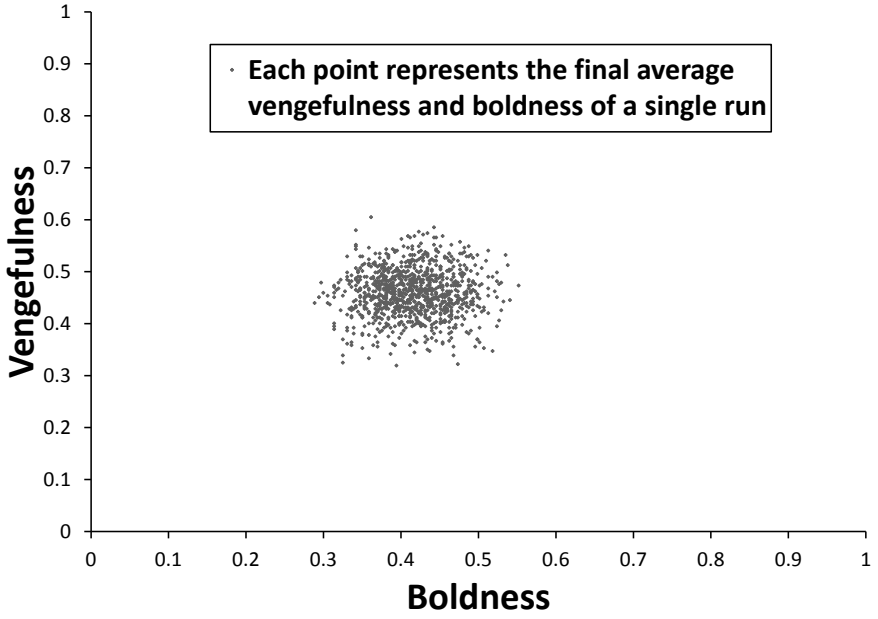


FIGURE 5.12: Scale-free network, 1,000,000 timesteps

$T = 3$, $E = -2$, $P = -9$ and $H = -1$. The results, shown in Figure 5.12, indicate that all runs end with both average boldness and average vengefulness of midrange values, so that no norm is established. However, by undertaking a detailed analysis of individual runs, it becomes clear that the reason for this is that there is no significant change to the average vengefulness and boldness, with both fluctuating around midrange value from the start of the run until the end.

By differentiating between hubs and outliers, some patterns are revealed, however. In particular, the model succeeds in lowering the boldness of hubs, but their vengefulness remains near the midrange. Because hubs are connected to many other agents and are thus punished many times for a defection, boldness decreases. Conversely, they also punish many of these other agents for defecting, and consequently pay a very high cumulative enforcement cost that causes them to lower their vengefulness. In turn, this lower vengefulness causes them subsequently not to punish others and as a result to receive metapunishment from other agents, leading to an increase in vengefulness

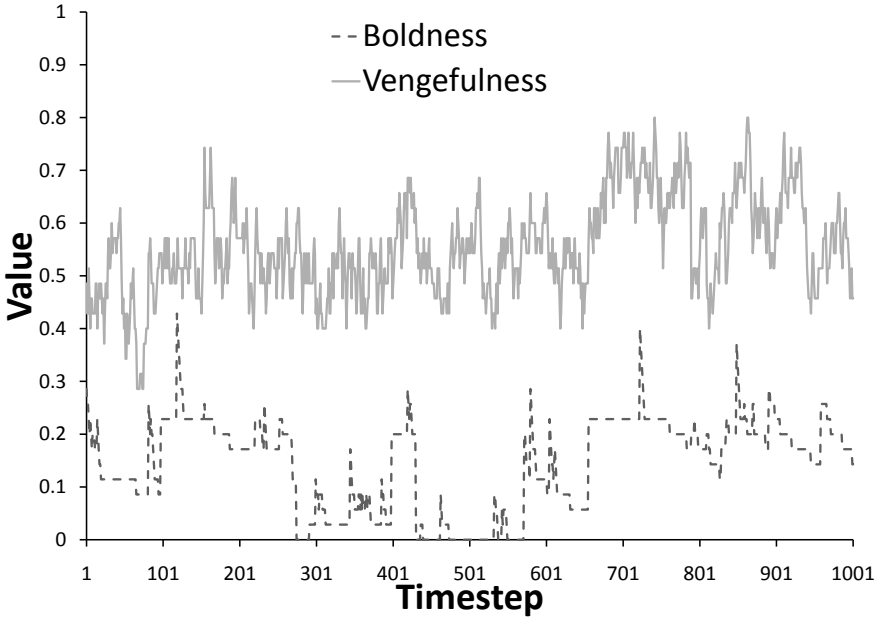


FIGURE 5.13: Hubs in scale-free networks

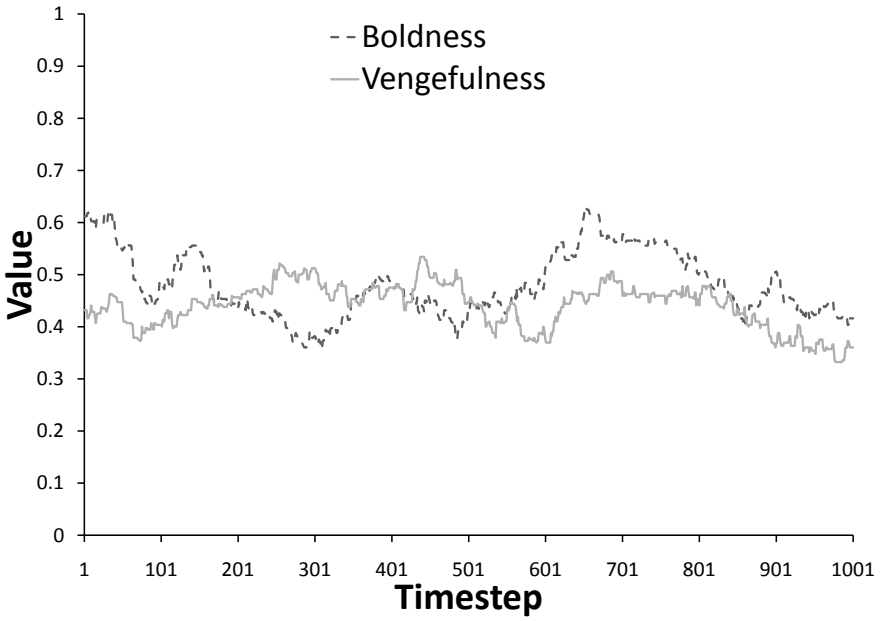


FIGURE 5.14: Outliers in scale-free networks

again. Over time, this repeats, with vengefulness decreasing and then increasing back to midrange, as shown in Figure 5.13. For the remaining, *outlier*, agents, changes to boldness and vengefulness are indicative of the overall boldness and vengefulness because they comprise the majority of the population. They are typically connected to one or more of the hubs, and while they too defect and punish, they do so much less frequently than the hubs to which they are connected. Thus, their scores are generally higher than the scores of the hubs; because those agents with higher scores do not learn from others (since there are no higher scoring others to learn from), they do not change their strategies, and their boldness and vengefulness remain close to the midrange value, as shown in Figure 5.14. These results demonstrate that our algorithm is not effective in scale-free networks. Importantly, as the burden of punishment falls largely on hubs rather than outliers, hubs perform worst in the population. To address this, we modify the learning technique so that it can cope with the nature of scale-free networks. The updated algorithm is discussed next.

5.5.1 Universal Learning

The algorithm proposed earlier suffers from the limitation that it requires knowledge of the average score in the population in order for an agent to determine whether to modify its policies. However, since our aim is to eliminate the unreasonable assumption of *omniscience*, by which agents are able to observe the private strategies of others, as well as observing all norm violations and punishments, it makes little sense to assume that agents have access to an average population score against which to compare themselves before deciding whether to modify their policies. For this reason, we consider here an alternative approach, in which agents always modify their policies to improve performance, regardless of the behaviour of others, and only in relation to their own

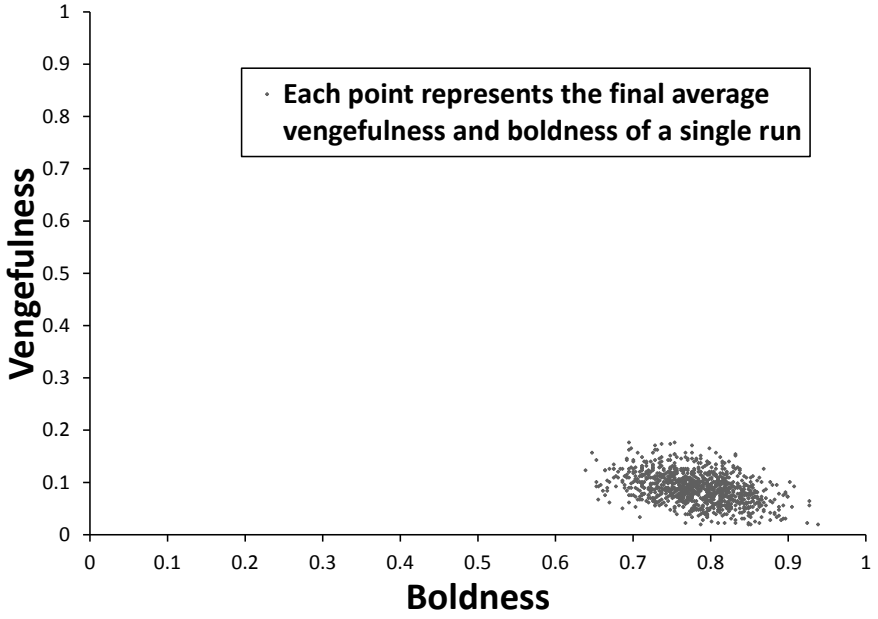


FIGURE 5.15: Universal learning, 1,000,000 timesteps

score. This modification is simple, and involves removing line 3 of Algorithm 6 as shown in Algorithm 7.

Experiments with this new approach give the results shown in Figure 5.15. Surprisingly, the results indicate norm collapse, as all runs end with high boldness and low vengefulness. By analysing the performance of the different types of agents, we are able to explain this behaviour; we illustrate by reference to a sample run for a hub in Figure 5.17, and a sample run for an outlier agent shown in Figure 5.16.

Outliers have few connections, but are connected to one or more hubs. When agents punish others, they pay an enforcement cost but risk metapunishment when they do not. However, since these outliers have very low connectivity, the risk of metapunishment is also very low, so they avoid punishing others and vengefulness consequently decreases. Metanorms are thus not effective here because of the lack of connectivity between agents. As a result, outliers always have high boldness and low vengefulness levels. In

Algorithm 7 learn(γ, δ)

```
1.  for each agent i do
2.    if explore(γ) then
3.      B_i = random()
4.      V_i = random()
5.    else
6.      if DS_i < 0 then
7.        if B_i − δ < 0 then
8.          B_i = 0
9.        else
10.         B_i = B_i − δ
11.     else
12.       if B_i + δ > 1 then
13.         B_i = 1
14.       else
15.         B_i = B_i + δ
16.     if PS_i < POS_i then
17.       if V_i − δ < 0 then
18.         V_i = 0
19.       else
20.         V_i = V_i − δ
21.     else
22.       if V_i + δ > 1 then
23.         V_i = 1
24.       else
25.         V_i = V_i + δ
```

addition, and as we will see, the vengefulness of hubs also drops and is never higher than midrange level, so agents can defect and gain the benefit of doing so, without being punished by hubs. Outliers thus increase their boldness, causing norm collapse in the whole population.

In contrast to outliers, hubs are highly connected and thus apply punishments to many others, incurring high enforcement costs. To address this, they decrease their vengefulness, resulting in metapunishment from the many nodes to which they are connected, in turn causing hubs to increase their vengefulness (but only to a mid-range level). In addition, because of the high boldness of outliers, there is a high rate of defection in the population, causing oscillation between mid-range and low vengefulness for the duration of the run. Boldness of hubs is kept at a low level, however, due to the amount

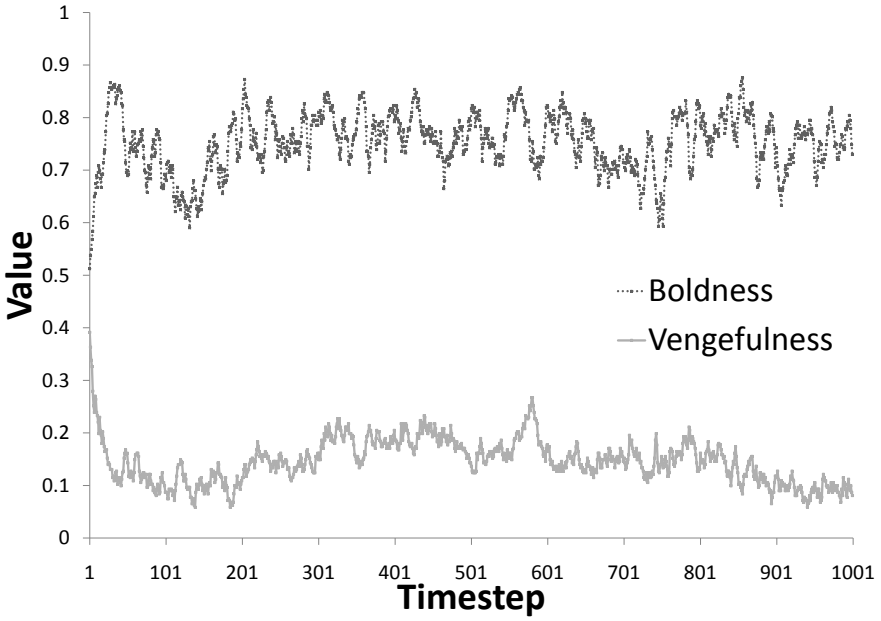


FIGURE 5.16: Universal learning: Outliers

of punishment that the hubs are exposed to.

5.5.2 Connection-Based Observation

Axelrod's original model considers a probability of being seen, and in the context of a fully connected network, this may be a reasonable basis on which to base a model. However, in the kinds of topologies we are concerned with, such as those that reflect the situations in peer-to-peer (P2P) networks or wireless sensor networks, for example, observation of the behaviour of others arises from the direct connection between agents. Thus, if a peer x is connected to another peer y, then x is able to observe all communication from y. As a result, if y defects by, for example, not sharing files in the case of a file-sharing P2P network, this is observed by x. To reflect this property in our model, Axelrod's probability of being seen requires replacing with the notion that

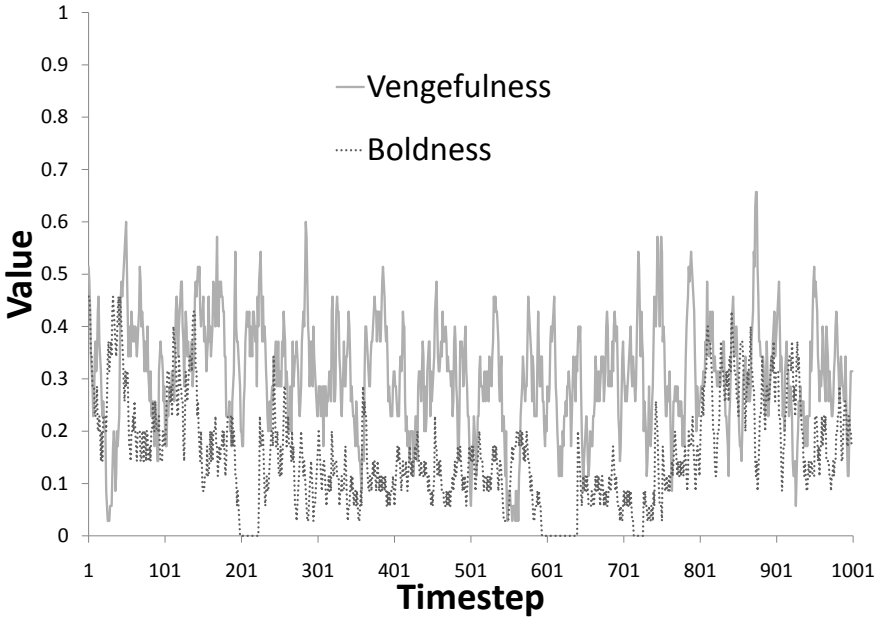


FIGURE 5.17: Universal learning: Hubs

each agent observes all actions of its direct neighbours. This modification to the model gives rise to rather different results.

In particular, the results of running the model on a scale-free network, in Figure 5.18, show that all runs end in low boldness and low vengefulness, indicating that defection is very rare in the population because of the low boldness. In addition, punishment is not common since agents rarely punish defectors, due to their low vengefulness. To understand this better, the results of the first 1,000 timestep of a sample run, for outliers and hubs, are shown in Figures 5.19 and 5.20, respectively.

More specifically, Figure 5.19 shows that outliers start the run by decreasing both vengefulness and boldness to a low level where they remain, with some small degree of fluctation. Figure 5.20 suggests that hubs start the run by increasing their vengefulness to a high level and decreasing their boldness to a very low level. After a few timesteps, vengefulness decreases to a mid-range level, from which it decreases further to a low

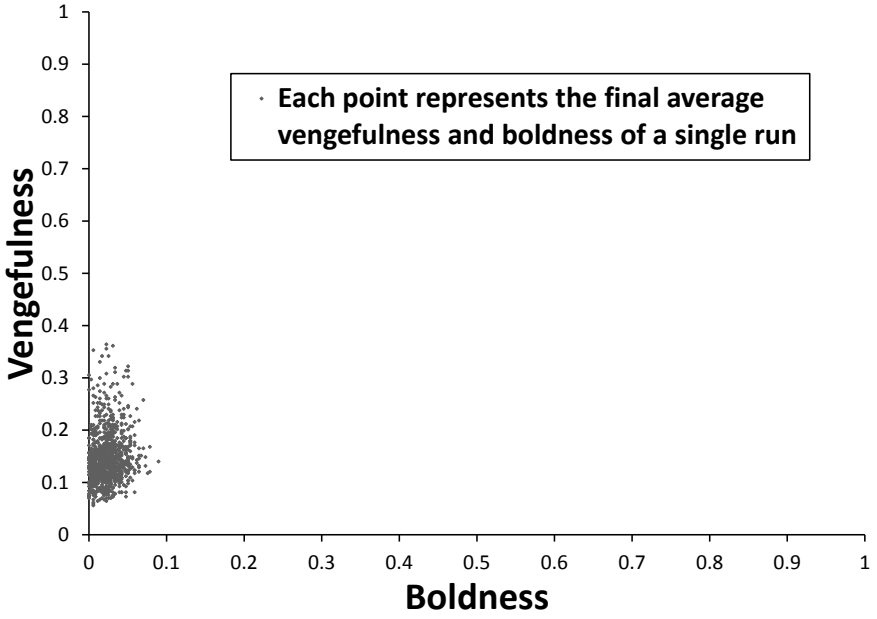


FIGURE 5.18: Connection-Based Observation, 1,000,000 timesteps

level. However, it does not stabilise there, since it moves up again, and this pattern is repeated throughout the run. Similarly, boldness initially decreases to zero and then jumps to a low level, before decreasing back to zero. Hubs thus have a fluctuating mid-range level of vengefulness, and a very low level of boldness.

There are two distinctive features that can be observed here, in contrast to the results obtained by the universal learning approach. First, hubs reach a high level of vengefulness, which is limited to mid-range vengefulness in the previous approach. This is mainly because the new technique raises the action observation probability to 100%, which allows a high possibility for metapunishment to occur and, as a result, forces hubs to increase their vengefulness to a high level. However, as before, this does not persist because of the high enforcement cost observed with such a high level of vengefulness. Second, the boldness of outliers is low here, mainly due to the combination of the high vengefulness among hubs and the 100% defection observation, which together produce sufficient punishments to force outliers to decrease their boldness.

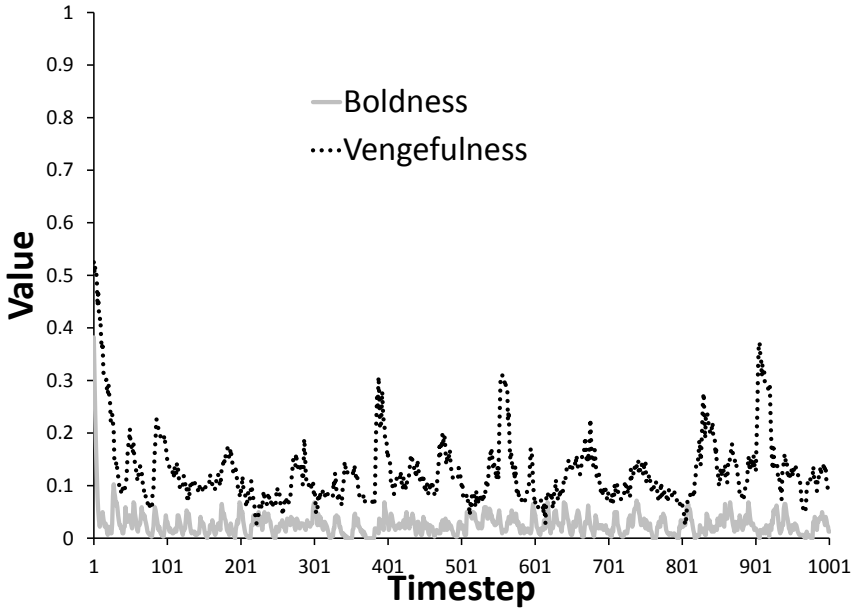


FIGURE 5.19: Connection-Based Observation: Outliers

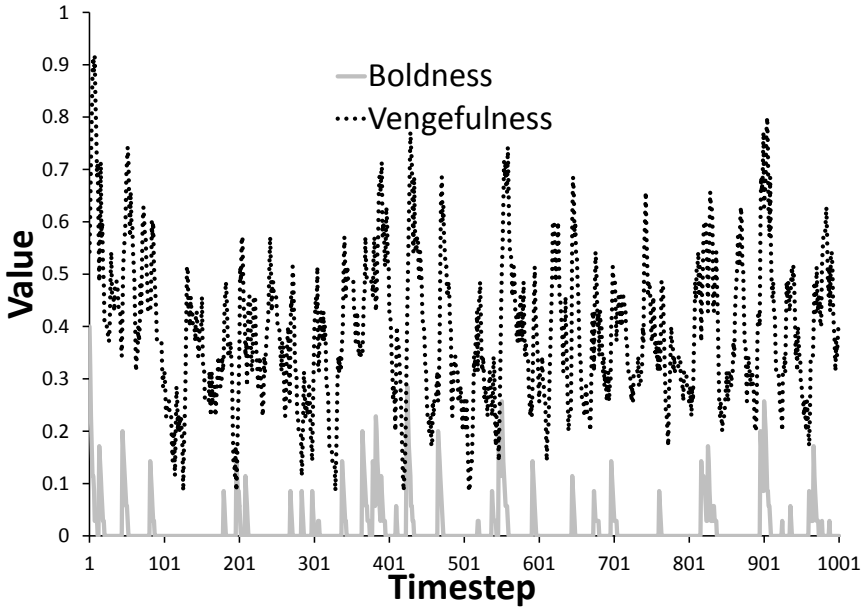


FIGURE 5.20: Connection-Based Observation: Hubs

5.6 Dynamic Policy Adaptation

As we have seen, universal learning has a negative impact on the results, causing boldness to increase and vengefulness to decrease. However, there is a more important weakness of the model in that the learning rate is uniform in the face of differing punishment levels. More specifically, all agents use the same learning rate, regardless of how much utility gain or loss they suffer. Thus, for example, an agent that incurs a punishment score of -10 must modify its vengefulness to exactly the same degree as another agent whose punishment score is -999. While the direction of change is appropriate, the degree of change does not reflect the severity of the sanction; a more appropriate approach would be to change policy in line with performance. In this view, a very badly performing agent should modify its policy much more significantly than one that does not perform as poorly. In this section, we consider dynamic policy adaptation to address this weakness, and to bring about changes to vengefulness and boldness that reflect performance.

The key notion underlying our technique is to measure the *level* of performance rather than just the *direction*. This can be achieved through comparison of an agent's actual utility or *score* in our terms, and the maximum or minimum that could be obtained. We apply this principle to boldness and vengefulness in turn. Before proceeding, we introduce some notation. Let NDD be the number of available defection decisions, where each agent can have more than one chance to defect in a single round (as specified earlier), $|NB_i|$ be the number of i's neighbours, T be the utility that can be gained from a single defection, and P be the punishment cost that represents the utility lost from being punished.

5.6.1 Boldness

In terms of boldness, the relevant part of the total score is the *defection score*, which can be either positive or negative, requiring consideration of both maximum and minimum possible values. The maximum possible defection score $MaxDS_i$ arises when an agent i always defects but is never punished, as follows.

$$MaxDS_i = NDD \times T \tag{5.1}$$

In contrast, the minimum defection score that can be obtained by an agent arises when the agent always defects and is always punished by all of its neighbours, as follows.

$$MinDS_i = NDD \times (T + (|NB_i| \times P)) \tag{5.2}$$

Then, in order to determine the degree of change to an agent i's boldness ($FactorB_i$, see Equation 5.3), we must consider three different situations. First, when the defection score is positive (so that boldness should increase), the degree of change is determined by dividing the obtained defection score by the maximum possible defection score. Second, when it is negative, (so that boldness should decrease), the obtained defection score is divided by the minimum possible defection score. Finally, if the defection score is zero, no change is required.

$$FactorB_i = \begin{cases} \frac{DS_i}{MaxDS_i} & \text{if } DS_i > 0 \\ \frac{DS_i}{MinDS_i} & \text{if } DS_i < 0 \\ 0 & \text{otherwise} \end{cases} \tag{5.3}$$

Given this, we now need to determine how $FactorB_i$ can be used to change agent i's policy. In order to avoid dramatic policy movements that could lead to violent

fluctuations, we limit the change that can be applied to a maximum value. In this case, the maximum value is the difference between two levels as in Axelrod's original model, of $\frac{1}{7}$. Thus, in terms of boldness, agent i modifies its boldness in line with its DS_i, as follows.

$$B_i = B_i + \begin{cases} \frac{1}{7} \times FactorB_i & \text{if } DS_i > 0 \\ -\frac{1}{7} \times FactorB_i & \text{if } DS_i < 0 \\ 0 & \text{otherwise} \end{cases} \tag{5.4}$$

This means that an agent can maximally change its boldness by one level (or by $\frac{1}{7}$) when *FactorB* is 1.

5.6.2 Vengefulness

An agent modifies its vengefulness depending on whether it is valuable to punish others, determined by comparing the utility lost from punishing others (the punishment score, *PS*) against the utility lost from not punishing them (the punishment omission score *POS*). If *PS* is worse than *POS*, agents decrease vengefulness and increase it otherwise. Clearly, the magnitude of this difference between these two values gives an indication of the degree of change that should be applied to vengefulness. For example, if *PS* is -24 and *POS* is -20, then the amount of decrease to V should be significantly lower than when *PS* is -600 and *POS* is -20. We call this difference *DiffV*:

$$DiffV_i = |PS_i - POS_i| \tag{5.5}$$

Since *DiffV* is 1 or more (when the values are not equal), it cannot be used directly to update an agent's V value, because V must always lie between 0 and 1. It must thus be *normalised* so that it can be applied to V, for which we use a scaled value, *FactorV*; this is determined by dividing *DiffV* by the minimum of *PS* and *POS*. Since both *PS*

and *POS* are negative, the absolute value of the minimum of these two scores is used for the scalling:

$$FactorV_i = \frac{DiffV_i}{|\min\{PS_i, POS_i\}|} \tag{5.6}$$

While this always produces a value between 0 and 1, it does not provide the same value for the same magnitude of difference. For example, if *PS* is -14 and *POS* is -20, we want $FactorV_i$ to be the same as when *PS* is 0 and *POS* is -6. We can achieve this by replacing $|\min\{PS_i, POS_i\}|$ with the maximum possible difference between *PS* and *POS*. This maximum difference is the difference from 0 (obtained when there is no cost at all from punishing or from not punishing) and the greatest possible magnitude of *PS* or *POS*. The highest punishment score *HPS* represents (the maximum in magnitude, and lowest in numerical terms — we use *HPS* to indicate the *highest* score to avoid ambiguity of minimum and maximum) is received by an agent punishing all of its neighbours for defection, and metapunishing all of its neighbours for not punishing all of their neighbours for defection.

To determine the value of *HPS*, we need to consider both the punishment enforcement cost and the metapunishment enforcement cost. First, the highest (maximum in magnitude, but minimum numerically) *punishment* enforcement cost (*HPEC*) arises when all of an agent's neighbours defect and the agent punishes all of them:

$$HPEC_i = NDD \times |NB_i| \times E \tag{5.7}$$

where E is the enforcement cost of a single punishment. Similarly, the highest *metapunishment* enforcement cost (*HMPEC*) arises when all of an agent's neighbours do not punish all of their neighbours for defecting, and the agent metapunishes all of them:

$$HMPEC_i = NDD \times |NBB_i| \times E \tag{5.8}$$

where $|NBB_i|$ is the total number of neighbours of all of agent i's neighbours.

Based on this, the highest punishment score of agent i is defined as follows:

$$HPS_i = HPEC_i + HMPEC_i \tag{5.9}$$

In the same way, the highest punishment omission score *HPOS* (greatest in magnitude, lowest numerically) can be obtained when an agent does not punish any defectors, but is metapunished by all of its neighbours, as follows:

$$HPOS_i = NDD \times |NB_i| \times (|NB_i| - 1) \times P \tag{5.10}$$

where the maximum number of defectors is all of an agent's neighbours ($|NB|$), the maximum number of metapunishers is the same but excluding the defecting agent, and P is the punishment cost obtained from being metapunished (which is the same as for simply being punished).

Given this, $FactorV_i$ of agent i can be calculated (see Equation 5.11) by dividing *DiffV* by one of these values, as follows. If punishing brings a greater utility reduction than not punishing ($PS < POS$), then we use the highest punishment score *HPS*. Conversely, if $PS > POS$, then we use the highest punishment omission score *HPOS*. If there is no difference, then there is no change and $FactorV$ is equal to 0.

$$FactorV_i = \begin{cases} \frac{DiffV_i}{|HPS_i|} & \text{if } POS_i > PS_i \\ \frac{DiffV_i}{|HPOS_i|} & \text{if } POS_i < PS_i \\ 0 & \text{otherwise} \end{cases} \tag{5.11}$$

This guarantees that the change made to V is always the same given the same difference in scores, since both *HPS* and *HPOS* are fixed for each agent. Moreover, this approach

allows hubs to change much less quickly than outliers, because the highest (maximum in magnitude) scores for hubs are much higher than for outliers, so that the results achieved by using $FactorV$, and dividing by the difference in scores obtained for hubs, is much less than for outliers.

According to $FactorV_i$, agent i thus increases vengefulness when it finds that not punishing is worse than punishing, and it decreases vengefulness when the converse is true, as follows:

$$V_i = V_i + \begin{cases} \frac{1}{7} \times FactorV_i & \text{if } PS_i > POS_i \\ -\frac{1}{7} \times FactorV_i & \text{if } PS_i < POS_i \\ 0 & \text{otherwise} \end{cases} \tag{5.12}$$

5.6.3 Example

To illustrate, assume that a hub x is connected to 20 other agents, and that an outlier y is connected to only 2 other agents (one being a hub). Like Axelrod's seminal experiments and without loss of generality, let $NDD = 4$ for all agents, since every agent has 4 chances to defect in each round. $E = -2$ and is the same for all agents. Similarly, $P = -9$ and again is the same for all agents. The temptation value for all agents, received when they defect, is $T = 3$. Finally, suppose that x's neighbours have 50 other distinct neighbours in total (summed over all neighbours), while y's neighbours have 20 other distinct neighbours (again, over all). This is summarised in Table 5.1. Given these values, we can determine the relevant values needed as follows. Starting with defection scores and from Equations 5.1 and 5.2 respectively, we obtain the following:

$$MaxDS_x = MaxDS_y = 4 \times 3 = 12$$

$$MinDS_x = 4 \times (3 + (20 \times -9)) = -708$$

$$MinDS_y = 4 \times (3 + (2 \times -9)) = -60$$

TABLE 5.1: Example values for Agents x and y

Agent	Pos	$\|NB\|$	*NBB*	*MinDS*	*MaxDS*	*LevB*	*HPS*	*HPOS*	*LevV*
x	Hub	20	50	-708	12	1/7	-560	-13680	1/7
y	outlier	2	20	-60	12	1/7	-176	-72	1/7

In terms of punishment scores, from Equations 5.7, 5.8 and 5.9, we obtain the following:

$$HPEC_x = 4 \times 20 \times -2 = -160$$

$$HMPEC_x = 4 \times 50 \times -2 = -400$$

$$HPS_x = -160 - 400 = -560$$

$$HPEC_y = 4 \times 2 \times -2 = -16$$

$$HMPEC_y = 4 \times 20 \times -2 = -160$$

$$HPS_y = -16 - 160 = -176$$

Punishment omission scores using Equation 5.10 are as follows:

$$HPOS_x = 4 \times 20 \times 19 \times -9 = -13680$$

$$HPOS_y = 4 \times 2 \times 1 \times -9 = -72$$

Using this information (Table 5.1), we can determine the decisions for specific situations. For example, at the start of each run, the population has midrange average boldness and vengefulness (because of the uniform distribution function to generate initial policies). Now, suppose that both x and y also have mid-range boldness and vengefulness. If, after one round, both x and y defected twice (out of their four opportunities to defect), they each gain twice the temptation value T. However, since x is a hub, suppose it is punished 22 times, much more than y, which is only punished twice. This is because the defection score of a hub with mid-range boldness is typically much worse than that of a similar outlier, mainly due to the difference in their number

of neighbours, and the midrange vengefulness in the population. Thus, x has a defection score of 2×3 from defecting, plus $22 \times -9 = -198$ from being punished, giving $DS_x = -192$. Similarly, $DS_y = ((2 \times 3) + (2 \times -9)) = -12$.

Given these defection scores, the degree of change that each agent applies to its boldness can be calculated as follows. First, from Equation 5.3, $FactorB_x = \frac{-192}{-708} = 0.3$ and $FactorB_y = \frac{-12}{-60} = 0.2$. Now, using Equation 5.4, and since both DS_x and DS_y are negative, B_x is decreased by $0.3 \times \frac{1}{7} = 0.04$, and B_y by $0.2 \times \frac{1}{7} = 0.03$.

In addition, if x punishes 20 other agents and metapunishes 10 more, and y punishes 2 other agents and metapunishes 1 more, their punishment scores are determined by multiplying the number of punishments issued by the enforcement cost E: $PS_x = ((20+10)\times-2) = -60$ and $PS_y = ((2+1)\times-2) = -6$. Then, if x has spared 10 defectors and has been metapunished 2 times for each instance of omitting punishment, and y has spared only one defector and been metapunished just once, the punishment omission scores are calculated by multiplying the number of metapunishments by the punishment cost P, as follows: $POS_x = (10 \times 2 \times -9) = -180$ and $POS_y = (1 \times 1 \times -9) = -9$. Thus, by Equation 5.11, $FactorV_x = \frac{|-60-(-180)|}{13680} = 0.01$ and $FactorV_y = \frac{|-6-(-9)|}{72} = 0.04$. Then, since $PS_x > POS_x$, x increases its vengefulness V_x by $0.1 \times \frac{1}{7} = 0.001$ according to Equation 5.12. Similarly, since $PS_y < POS_y$, y decreases its vengefulness by $0.04 \times \frac{1}{7} = 0.006$.

5.6.4 Experimental Results

To determine the effect of introducing dynamic policy adaptation, we ran experiments, similar to the previous experiments, on the new model, and giving the results shown in Figure 5.21. As can be seen from the figure, all runs result in populations with low average boldness and moderate vengefulness. As before, more detail on the evolution

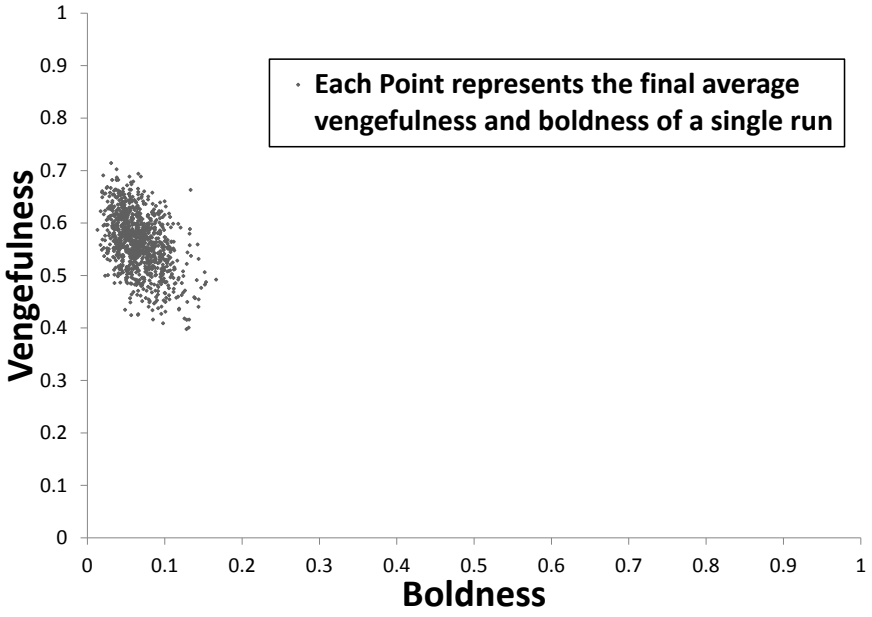


FIGURE 5.21: Dynamic Policy Adaptation, 1,000,000 timesteps

of average boldness and vengefulness for hubs and outliers was provided by examining runs of individual agents, as shown in Figures 5.22 and 5.23, which confirm that outliers converge to a state of low boldness and moderate vengefulness consistently, while hubs do so with intermittent deviations.

As before, hubs increase vengefulness and decrease boldness, though much more slowly in the new model. However, at regular points in time, there are sudden decreases to vengefulness, accompanied by sudden increases in boldness, as a result of the exploration component of the algorithm.

5.7 Conclusion

In this chapter, we have investigated mechanisms that encourage norms to emerge in communities of self-interested agents, without interference of a central or outside

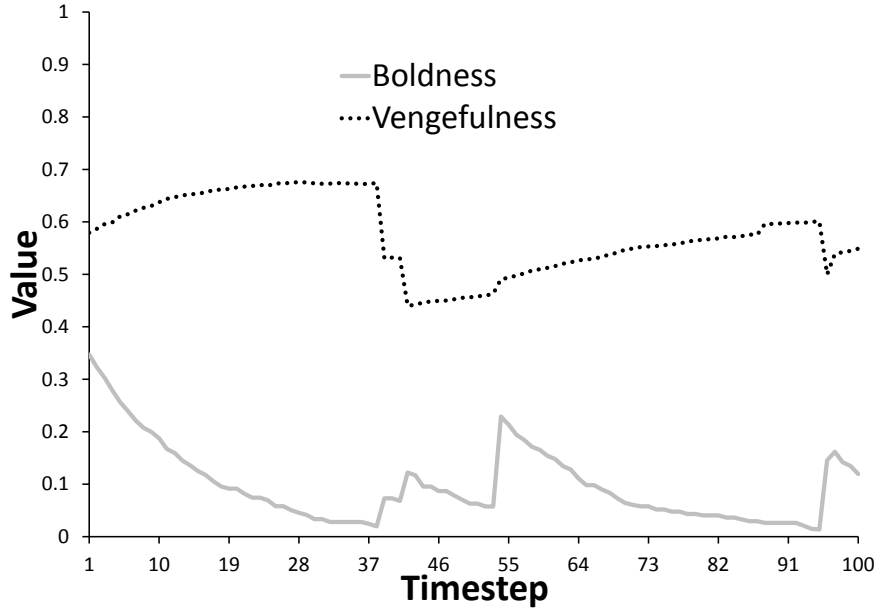


FIGURE 5.22: Dynamic Policy Adaptation for Hubs

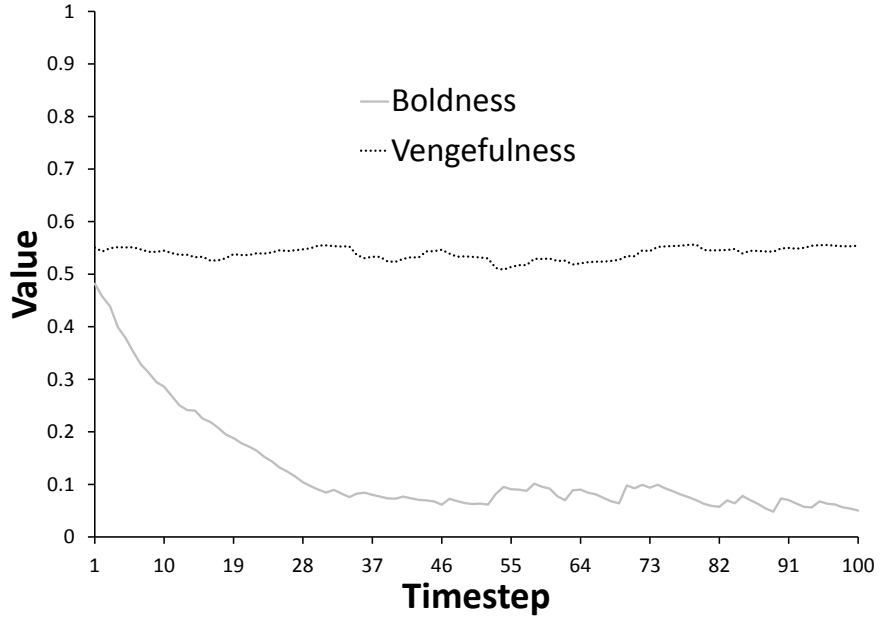


FIGURE 5.23: Dynamic Policy Adaptation for Outliers

authority, under the realistic constraint that agents can only influence one another if they regularly interact. Based on Axelrod's seminal work, our model's substantial novel extension examines the impact of different types of topologies of interaction on norm emergence. Our results show that in circumstances in which *each* agent regularly interacts with a small number of other agents, as in lattices and small worlds, our mechanisms to encourage norm emergence remain largely effective. More precisely, it is very effective for lattices, but its effectiveness varies with the rewiring probability in small worlds. Moreover, we have demonstrated that, given fixed penalties, for lattices, the effectiveness of the proposed approach only depends on the number of neighbours of each agent, *not* on the total population size. For small worlds, increasing the population size with a high rewiring probability decreases vengefulness, constraining norm emergence significantly. Thus, topology must be considered: in the case of a lattice or a small world, the proposed approach will be effective for sufficiently large neighbourhood sizes.

In terms of scale-free networks, our results show that the basic developed algorithm does not work. Our simulations suggest that poorly connected agents receive little discouragement from defecting while hubs are discouraged from enforcing norms through high enforcement costs. In response, we have modified the experimental setting to be more consistent with the nature of distributed systems of partially connected nodes, bringing an even more serious breakdown in norm emergence, but also subsequently addressed this through a dynamic policy adaptation mechanism. In this way, agents are able to change their policy in proportion to the punishments scores they receive, allowing them to adapt proportionally, and to maintain the policy values that sustain norm establishment.

Chapter 6

Efficient Norm Emergence through Adaptive Punishment

6.1 Introduction

Researchers from many scientific areas have considered *punishment* as a key motivating element for norms to be established [37, 43, 56, 139]. Therefore, several different punishment schemes to address these concerns have been considered [30, 49, 60], but all are *static* in that they consist of applying the same fixed penalty regardless of the circumstances. As such, they rely on the system developer to find an appropriate fixed penalty, yet the consequences of employing an inappropriate penalty are considerable. If it is too weak, punishment may be ineffective as a means of discouraging behaviour that does not support norm establishment. If it is excessive, it risks undercutting the benefits of cooperation. For example, in peer-to-peer file sharing systems, ostracising non-compliant individuals for extended periods of time may encourage norm compliance in the long run, but at the expense of losing the contributions of those individuals.

In response, this Chapter is concerned with how *adaptive* punishment may help a norm to emerge in a population of agents that starts out without any norm recognition. More specifically, the chapter addresses the problem of determining the punishment most appropriate to the violation context, by providing an adaptive punishment technique. In order to evaluate this adaptive punishment approach, it is integrated with the models that have been developed in previous chapters. While the notion of dynamic sanctioning, by which the level of punishment is modified in response to circumstances, has been suggested by Villatoro et al. [139], they consider only global modifications for a population as a whole rather than dynamism in relation to a particular agent's circumstances, as we do here.

The key contribution of the chapter is a novel technique for dynamically adapting punishments to suit the circumstances of individual agents so that an overall system is more efficient. This is extensively evaluated with a series of experiments. The rest of the chapter is structured as follows. We continue in the next section by introducing the necessary changes to the metanorm model in the light of the adaptive punishment approach. Then in Section 6.3, we introduce our experiential adaptive punishment approach and evaluate the model with analysis of extensive experimental results concerned with different aspects of our approach. Section 6.4 describes the adaptation of the model to facilitate pairwise interactions at a distance, before concluding in Section 6.5.

6.2 Metanorms and Adaptive Punishment

In seeking to develop a model capable of supporting norm emergence in real world distributed systems, we integrate the adaptive punishment approach, with the metanorm model as a means of studying norm emergence. In what follows, we first introduce the

concept of adaptive punishment and compare it to static punishment. After that, we introduce the changes that have been applied to the model in order to integrate the adaptive punishment approach.

6.2.1 Adaptive and Static Punishment

Consider again the example of peer-to-peer (P2P) file sharing mentioned in the introduction, in which agents are able to download files from each other. Here, there is a norm obliging agents participating in the system to upload the files they have downloaded in order to share them with others, and to maintain the availability of these files on the network. However, since uploading consumes bandwidth, and in the absence of an appropriate punishment, self interested agents could choose not to share (upload) files they have downloaded in order to preserve bandwidth for their own use. In the case of frequent occurrences of such selfish behaviour, the efficiency of the entire P2P network can be threatened. In response to this problem, de Pinninck et al. [29] suggest that the punishment for such behaviour should be *blocking*, by which all other agents cease interacting with an agent that is observed not to share files after downloading them.

If punishment is static, it can be pre-specified at design time. In the P2P example, such static punishment involves identifying agents that violate the norm, and blocking them for the specified duration, on each occurrence of a violation. While this fixed blocking period is determined by the system designer, the challenge is to determine the most appropriate blocking period. For example, if the blocking period is fixed at 30 minutes for each violation, this may be just the right penalty needed to regulate the behaviour of some agents, but others may be persuaded to avoid violation (and upload files as demanded by the norm) with a shorter blocking period of only 10 minutes. Such a situation suggests that the network is losing the potentially valuable

participation of some agents for a period of 20 minutes. Conversely, there may be still more agents for whom 30 minutes is insufficient to convince them to cease violations, for example due to the gain from violation exceeding the loss incurred through being blocked for 30 minutes. It is in this sense that determining an appropriate blocking period can be crucial to the performance of the overall system, especially those that rely on the participation of members for their functionality and effectiveness (as with P2P networks). However, determining such an appropriate blocking period can be difficult, if not impossible, simply because there is no single fixed period that is effective when dealing with different types of agents.

In consequence, adaptive punishment can play an important role in overall system performance by adapting punishment values in line with available information about the violating agent, so that punishment is increased when the current value is found to be ineffective, and decreased when it is excessive. Returning to our example, if the agent does not take the opportunity to share files at the first opportunity after the blocking period has ended, then it can be blocked again, this time for a longer period, with the aim of bringing about compliant behaviour. This can continue until the agent begins to comply (at least occasionally), when the duration of blocking for subsequent infrequent violations may now be reduced.

Clearly, as the example above suggests, the appropriate punishment at any moment should be determined by an agent's prior behaviour: an agent that has a history of violation should be punished more than one with a history of compliance, in order to bring about a change to ingrained behaviour. Now, since agents themselves must apply punishments, they must maintain a record of their prior interactions (involving requests for files and decisions about whether or not to share these files) with others and build up a repository of experience to determine the level of punishment to apply.

6.2.2 History-Based Adaptive Policy Learning

As stated in Section 5.6, agents do not adapt their policies in the same way: a policy that results in a very low utility is altered differently to a policy that is not as bad. Therefore, agents change their policies proportionally to their success, following the WoLF philosophy [19], so that if the utility lost from taking a certain action is high, then the change to the corresponding policy is high, and if the utility lost is low then the change to the policy is low.

The policy adaptation approach presented here is an enhanced version of the approach presented in Section 5.6, which depends on extreme cases (the best or worst scores that an agent can *possibly* obtain), which might not actually occur in real settings. In addition, it also depends on agents being able to estimate such best or worst scores that they may obtain in a single round. This is possible with static punishment, since agents know in advance the amount of punishment for a defection, which allows them to calculate the best and worst possible scores as described previously. However, such a calculation becomes much more complicated with the introduction of variable punishment, due to agents being unable to predict the amount of punishment and, as a result, not able to calculate the best and worst possible scores. In consequence, the approach needs modification. In this section we introduce these modifications, allowing agents to change their policies according to their recorded history, and more specifically according to the difference between their current utility and the best or worst utility obtained in their history of interactions.

First, the required change to an agent's boldness is calculated using the BAdaptiveLearning function of Algorithm 9, called by Algorithm 8. Here, agent i keeps track of two boldness-related historical variables: $HMaxDS_i$, which is the maximum obtained defection score in i's history of interaction, and $HMinDS_i$ is the minimum obtained defection score in i's history of interaction. These two variables are updated according

Algorithm 8 learn()

```
1. for each agent i do
2.    if explore(γ) then
3.       B_i = random()
4.       V_i = random()
5.    else
6.       BAdaptiveLearning(i, DS_i, oneLevel)
7.       VAdaptiveLearning(i, PS_i, POS_i, oneLevel)
```

Algorithm 9 BAdaptiveLearning($i, DS_i, oneLevel$)

```
1. if DS_i < 0 then
2.    HMinDS_i = min(HMinDS_i, DS_i)
3.    factorB_i = DS_i / HMinDS_i
4. else
5.    if DS_i > 0 then
6.       HMaxDS_i = max(HMaxDS_i, DS_i)
7.       factorB_i = DS_i / HMaxDS_i
8.    else
9.       factorB_i = 0
10. δB_i = oneLevel × factorB_i
11. return |δB_i|
```

to the current obtained defection score DS_i (See lines 2 and 6 of Algorithm 9). Then, $factorB_i$, which determines the change that should be made to agent i's boldness, is calculated based on the division of DS_i by $HMaxDS_i$ if DS_i is greater than zero, or on the division of DS_i by $HMinDS_i$ if DS_i is negative.

Given this, we now need to determine how $factorB_i$ can be used to change an agent's policy. Again, in order to avoid dramatic policy movements that could lead to violent fluctuations, we limit the change that can be applied to a maximum value, which is the difference between levels as in Axelrod's original model, of $\frac{1}{7}$ (represented by *oneLevel* in the algorithms). Thus, the modification to boldness, δB_i, can be calculated as shown in Line 10 of Algorithm 9. Note that when $factorB_i = 1$, then $\delta B_i = oneLevel = \frac{1}{7}$ (i.e. the maximum amount of change).

Second, with relation to vengefulness (as in the VAdaptiveLearning function of Algorithm 10, called by Algorithm 8), agent i keeps track of two historical variables:

Algorithm 10 VAdaptiveLearning($i, PS_i, POS_i, oneLevel$)

1. $differV_i = |PS_i - POS_i|$
2. $HMinPS_i = min(HMinPS_i, PS_i)$
3. $HMinPOS_i = min(HMinPOS_i, POS_i)$
4. **if** $POS_i > PS_i$ **then**
5. $\quad factorV_i = differV_i / HMinPS_i$
6. **else**
7. $\quad$ **if** $POS_i < PS_i$ **then**
8. $\quad\quad factorV_i = differV_i / HMinPOS_i$
9. $\quad$ **else**
10. $\quad\quad factorV_i = 0$
11. $\delta V_i = oneLevel \times factorV_i$
12. **return** $|\delta V_i|$

$HMinPS_i$, which is the minimum obtained punishment score in i's history of interactions; and $HMinPOS_i$, which is is the minimum obtained punishment omission score in i's history of interactions. Having the current obtained punishment score PS_i and punishment omission score POS_i, agent i updates the historical variables accordingly (Lines 2 and 3 of Algorithm 10). Then, $factorV_i$, which determines the change that should be made to agent i's vengefulness, is calculated based on the division of $differV_i$ (the difference between PS_i and POS_i) by $HMinPS_i$ if PS_i is better than POS_i or on the division of $differV_i$ by $HMinPOS_i$ if PS_i is better than POS_i.

The very first change to an agent's policy is always the maximum possible change, because there is no historical data, and the current scores will be considered as maximum or minimum scores for later steps. However, these maximum or minimum scores can change if the agent obtains better or worse scores in later interactions.

6.3 Experiential Adaptive Punishment

In this section, we present the experiential adaptive punishment approach by which agents depend on their experience to determine the amount of punishment they are responding with to a norm violator. Such experience is mainly built through repeated

interactions, which allow agents to build certain images of other agents. These images can then be used to identify a suitable amount of punishment. In what follows, we first introduce the mechanism by which agents build their experience. Then, we introduce how such experience can be used to build the adaptive punishment decision. Finally, the evaluation of the experiential adaptive punishment is introduced.

6.3.1 Recording Experience of Violation

In the P2P file sharing example, two things must be recorded by each agent involved in any interaction in which one agent requests a file and the other decides whether to share it (cooperating, and thus complying with the norm) or not (defecting, and thus violating the norm): the identity of the other agent involved in the interaction; and the type of action taken by this other agent, whether cooperating or defecting. Each agent can thus build up a repository of records of interactions in this way over time, providing a store of information on which to base punishment decisions.

A limitless repository, however, can cause problems. Since our aim is to encourage norm-compliant behaviour in individual agents, and since the mechanism we propose seeks to amplify punishments in the case of repeat violations leading to bad experience, clearly a key target is to modify behaviour of such repeat offenders. Yet if we consider only the average prior behaviour of such repeat offenders over a long period, any recent adjustments towards compliance may be vastly outweighed in the repository by the long history of violations, bringing further increased punishment rather than the reduction in punishment that is warranted. In order to address this, therefore, we need some means to determine punishments in light of more recent interactions between agents rather than much older interactions that do not reflect current reality: agents that start to act in support of societal norms should not be punished severely just because they had previously behaved badly. A *window* over the repository, with a particular *window*

size, can thus be used to limit the interactions that are considered to a specific period of recent time, allowing agents to forget old violations and adapt their punishments to changes in the behaviour of others much more quickly and effectively. In this way, the prior defections of an agent that defected regularly in the past, but that has recently begun to cooperate, will be weighted much less in comparison to more recent compliance when determining the punishment value.

We define the *memory*, M_i, of an agent i, to be a set of cells, each containing the identifier $agID$ of the other agent, j, involved in an interaction and the action act taken by that agent in the interaction:

$$M_i = \{m_1, m_2, .., m_n\}$$

where n is the window size, and m_j is the jth cell:

$$m_j = \langle agID, act \rangle$$

6.3.2 Adaptive Punishment

Given this notion of memory, and within the available window of data, we can specify two useful measures from agent i's perspective (from its memory): the number of previous instances of defection of agent j (nd_j), and the number of previous instances of compliance of j (nc_j). In turn, this gives the *defection proportion*, dp_j, as follows:

$$dp_j = \frac{nd_j}{nd_j + nc_j}$$

This alone is not enough to determine the level of punishment, since the absolute number of defections is also relevant. An agent that violates a norm ten times merits a

greater sanction than one that violates it just once. Moreover, an agent that violates a norm once in ten instances merits a lower sanction than one that violates it ten times from 100 opportunities, since this indicates persistent and repeated offence. We reflect these concerns in what we call the local defection view of agent i on agent j, as follows.

$$LocalView : AGENT \times AGENT \rightarrow \mathbb{R}$$
$$\forall ag_i, ag_j \in AGENT : LocalView(ag_i, ag_j) = dp_j \times nd_j$$

where ag_i is the punishing agent; ag_j is the defecting agent; dp_j is the defection proportion of agent j in agent i's memory; and nd_j is the number of defections of agent j in agent i's memory.

Now, while we are interested in modifying punishments to suit the circumstances, we need an initial *punishment unit* (pu) as a basis for such modification. In this way, an applied punishment can be determined by multiplying the defection proportion with the absolute number of defections and the punishment unit. Punishment can thus be seen as a function that takes two agents and returns the punishment value applied by the first agent to the second:

$$ExpPunish : AGENT \times AGENT \rightarrow \mathbb{R}$$
$$\forall ag_i, ag_j \in AGENT :$$
$$ExpPunish(ag_i, ag_j) = LocalView(ag_i, ag_j) \times pu$$

Since, non-compliance with both norms and metanorms (not punishing a norm defector) is considered to be defection, each agent must make two punishment decisions: whether to punish a norm defector and whether to metapunish an agent that does not itself punish a defector.

The metapunishment decision is slightly different to the punishment decision, and depends on the number of previously unpunished (or *spared*) defectors by agent j (nds_j), and the number of previously punished defectors by agent j (ndp_j), giving a *sparing proportion*, sp_j:

$$sp_j = \frac{nds_j}{nds_j + ndp_j}$$

6.3.3 Evaluation

Based on the model just described, we carried out several experiments to understand the impact and potential of adaptive punishment. Importantly, all of our experiments are undertaken over a scale-free network (generated with Barabasi's algorithm [9]) since this has shown to be the most challenging (See Chapter 5). Our experiments were run over 1,000,000 time steps, with 1,000 agents, and with a parameter set-up illustrated in Table 6.1 .

TABLE 6.1: Parameter initialisation

Term	**Description**	**Value**
i, j	Individuals	A number to identify individual agents
B_i	Boldness of i	Uniform distribution from 0 to 1
V_i	Vengefulness of i	Uniform distribution from 0 to 1
T	Player's temptation to defect	+3
H	Hurt suffered by others as a result of an agent's defection	-1
pu	Punishment unit	-9 or -1
$oneLevel$	Maximum amount of change	$\frac{1}{7}$
γ	*exploration rate*	0.01

6.3.3.1 Adaptive Punishment Experiments

The results obtained from the introduction of the dynamic policy learning before integrating adaptive punishment, are shown in Section 5.6.4, indicating that the model is

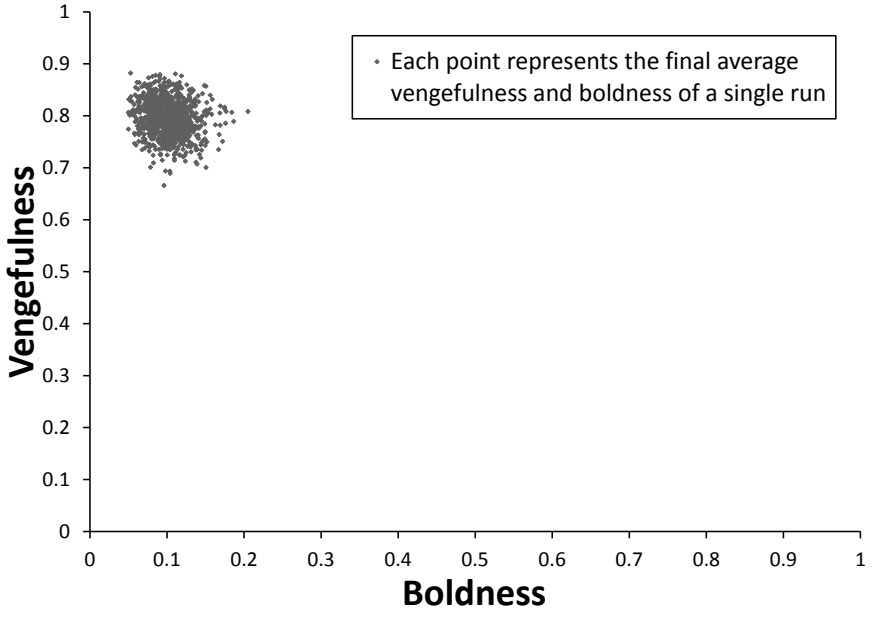


FIGURE 6.1: Experiential adaptive punishment results

successful to some degree in achieving norm emergence in an agent population over a scale-free network. While the approach is able to regulate agent behaviour with regard to norm defection (which can be seen from the near 0 average boldness of the population), the level of average vengefulness achieved is between 0.4 and 0.7, suggesting that norm emergence is good but not optimal; an agent can still defect without punishment.

The punishments that agents apply to each other are thus sufficient to learn that high boldness is harmful and should be reduced. However, while metapunishments are sufficient to prevent vengefulness from dropping below the midrange level, they do not cause agents to increase vengefulness adequately. We will not dwell further on these results, since a more detailed analysis of this situation has been provided in Chapter 5, but the important point to note is that the static punishment unit used here is -9 which, as will be shown later, is unnecessary and excessive.

By introducing adaptive punishment as described in this chapter, however, we are

able to improve the results as shown in Figure 6.1, in which the level of vengefulness has increased to be between 0.7 and 0.9. This is much more stable than previously, with high probability of punishing defectors and metapunishing those agents that do not punish defectors. The improvement can be explained by the flexibility brought about by adaptive punishment, since if a previous metapunishment does not succeed in changing behaviour, the metapunishment cost can be increased. Consequently, agents are encouraged to increase their vengefulness and punish norm defectors in the future. A more detailed analysis of the effect of this adaptive approach to punishment follows.

6.3.3.2 Punishment and Metapunishment Costs

While we can see that results improve through adaptive punishment, punishment and metapunishment values change in different ways depending on the situation. Figure 6.2 illustrates how two vengeful agents, a hub and an outlier (that punish regularly), adapt punishment when applying it to hubs and outliers with boldness and high boldness. In addition, for comparison, the figure includes a representation of the static punishment case of -9 that is used in the original model (shown as the horizontal line). Note that the x-axis represents the number of occurrences of punishment, rather than time-steps, so that each line indicates merely the trend of punishment applied by one agent to another.

It can be seen from Figure 6.2 that bold agents incur high punishments regardless of whether the punishing agent is a vengeful hub or a vengeful outlier. However, in the case of the bold hub, the maximum value of punishment is much less than in the case of the bold outlier (about 23 compared to 54). This is because, since a hub is more exposed (and has very many connections), it will be punished by many other agents, and thus punishment that is applied to a hub needs not be as high as for an outlier. Conversely, an outlier has few connections, requiring the punishment of a

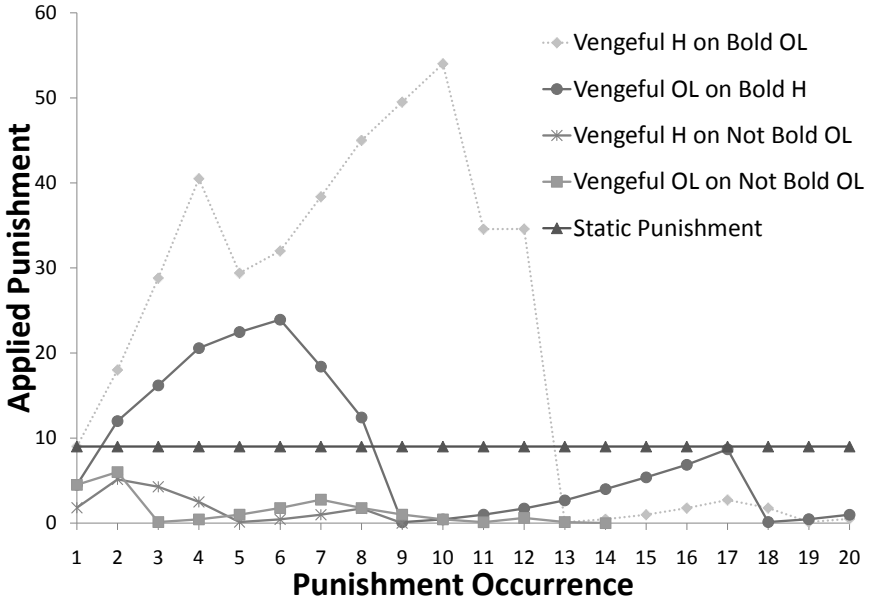


FIGURE 6.2: Punishment value v.s. punishment occurrence: $pu = -9$ (H: Hub; OL: Outlier)

single agent to be sufficiently high to convince it to stop defecting. In addition, it can be seen that the high punishment does not persist for many occurrences since it drops to a very low level after very few occurrences, because bold agents respond positively to the punishment and the high level ceases to be required. In this way, occasional stricter punishments bring about a relatively quick response when compared to the static approach, but demand a cumulatively lower amount of punishment, suggesting that adaptive punishment can be more efficient.

Interestingly, Figure 6.2 also shows that when a vengeful agent interacts with a low boldness agent, both the vengeful hub and the vengeful outlier always apply less punishment than in the static approach. This is because a low boldness outlier rarely defects and little is needed to prevent it defecting. Overall, it seems clear that the punishment cost here is much less than in the static approach, especially over an extended period (the duration of the simulation).

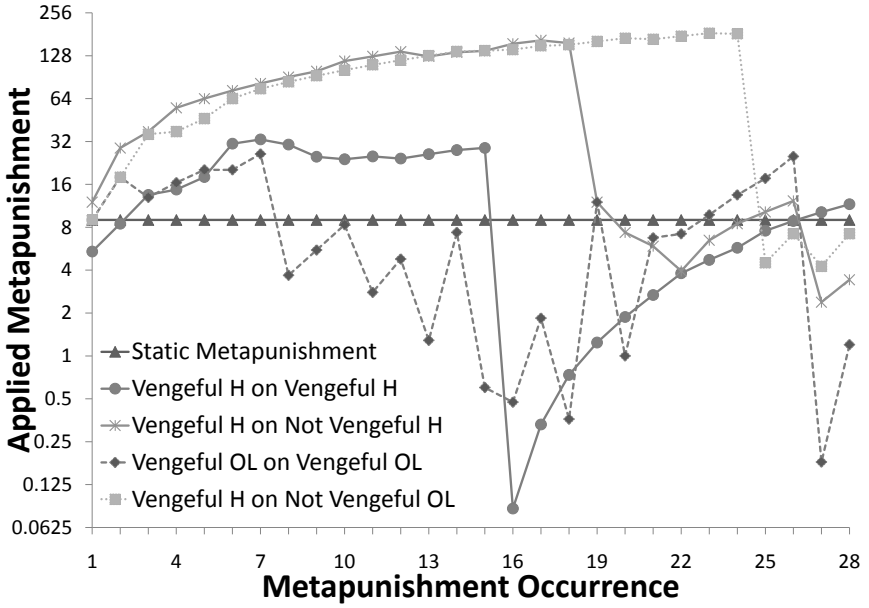


FIGURE 6.3: Metapunishment value v.s. punishment occurrence: $pu = -9$ (H: Hub; OL: Outlier)

Since punishment counters the tendency to defect, it needs to outweigh the impact of the temptation value (the reward from defecting). Importantly, an agent gains only one instance of this temptation value as reward, but can incur multiple punishments (from each agent to which it is connected) in response. In contrast, because metapunishment instead counters the enforcement cost, instances of which are incurred for each agent that should be punished or metapunished, this can lead to much higher metapunishment, as shown in Figure 6.3. Enforcement costs cause agents to decrease their vengefulness because they lose utility, but metapunishment seeks to balance this threat by losing even more utility from not punishing. The results in Figure 6.3 show that in our experiments, the metapunishment cost peaks at a level of 180, even though it is less than 9 (the fixed static metapunishment) in the majority of cases.

In fact, the different lines shown on the graph can be divided into two main categories: highly vengeful agents metapunishing weakly vengeful agents, and highly vengeful

agents metapunishing other highly vengeful agents. With regard to the first category, it is clear that very high metapunishments are needed to deal with agents that have low vengefulness. This is because such agents are less likely to impose punishments, especially given the enforcement costs that they are required to pay by increasing their vengefulness and punishing more agents as a result. However, such high metapunishment does not persist for long: it increases until around the 20th occurrence in the case of a hub, and until the 25th occurrence for an outlier. Then, metapunishment drops and remains below the traditional level of the static metapunishment.

With regard to the second category of highly vengeful agents, metapunishment only rises to a peak that is much lower than that of weakly vengeful agents. Moreover, within this, a high level of metapunishment persists much longer for the vengeful hub than for the vengeful outlier. This is because hubs have many more connections than outliers, so they are responsible for much more punishments and incur higher enforcement costs. In consequence, metapunishment needs to be much higher in order to counter this and to prevent the vengeful hub from decreasing its vengefulness to a low level too. While the same is true for vengeful outliers, it is for a much shorter period and for a lower cost, since outliers have far fewer connections.

6.3.3.3 Impact of Punishment Unit

In the previous experiments, a punishment unit of -9 was used, resulting in relatively high punishment. This is because, when a bold agent engages in a sequence of defections, this relatively high punishment unit is multiplied correspondingly, resulting in high punishment. In seeking to further understand the effect of the applied punishment unit, we undertook several experiments varying such unit between -9 and -1. Figure 6.4 shows the results of using a punishment unit of -1 (all other values give very similar results). From the figure, it seems clear that even a punishment unit of

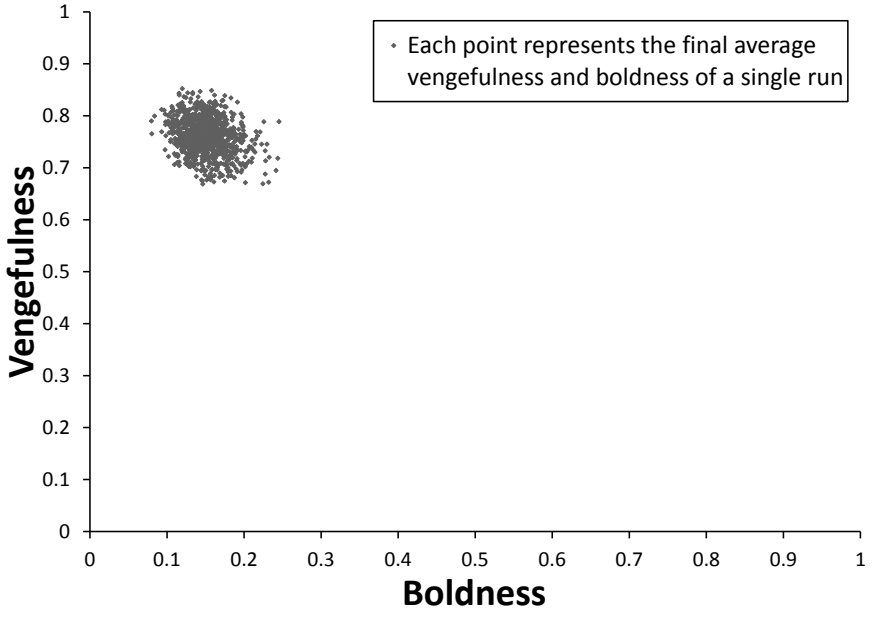


FIGURE 6.4: Impact of punishment unit when $pu = -1$

-1 is sufficient to achieve norm emergence, since the average level of boldness is still low and the average level of vengefulness is high.

However, a deeper analysis reveals that establishment of the norm takes slightly longer with the use of lower punishment units. This is due to the cumulative effect of adaptive metapunishment requiring a longer period to overcome the high cumulative enforcement cost, especially in the case of hubs. Yet, smaller punishment units ensure that much more appropriate levels of punishment are applied. For example, while a punishment of -4 can be sufficient to disincentivise an agent from defecting, such an actual level of punishment is unlikely to arise when a high punishment unit, such as -9, is used since it takes too long to reduce to that level. Note that only the adaptive punishment approach is able to achieve norm emergence when a low punishment unit is used. For example, the static punishment model does not succeed in achieving norm emergence if the static punishment cost is -1.

From Figure 6.5, which shows the change in punishments values over different occurrences when a unit of -1 is used, it is clear that in the vast majority of cases, the punishment applied does not exceed the fixed static punishment used in the original model. In fact, punishment exceeds 9 (but only going as high as 12) only in the difficult case of bold outliers, which have few connections and thus require much more effort to regulate their behaviour.

In terms of metapunishment, as shown in Figure 6.6, almost the same trend is observed when using a punishment unit of -1, but with some important differences. First, the maximum metapunishment here is 60 (not shown in the figure due to clarity reasons), as opposed to 180 in the case of punishment unit of -9 (see Figure 6.3). This is because, due to the lower punishment unit here, a much lower multiplier is brought into the equation. However, it requires about 140 metapunishment occurrences (not shown for same previous reasons) to regulate a weakly vengeful agent, compared to only 25 occurrences previously. Overall, much less metapunishment can be used to regulate agent behaviour than with higher punishment units, and with static punishment.

6.3.3.4 Emergent Agreement of Punishment

Based on the analysis of the results obtained from using the adaptive punishment approach with a variation of basic punishment units, there is a very interesting characteristic observed in relation to the agreement of punishment value. The agreement of punishment value means that all agents that decide to punish a defector do so with exactly the same amount. For example, if an agent has three different neighbours and the agent has defected twice already. According to the model, all the neighbours would have observed the previous defections and would have recorded them in their memory regardless of whether they have punished the defecting agent for these defections or not. Because all agents are following the same technique for calculating punishment,

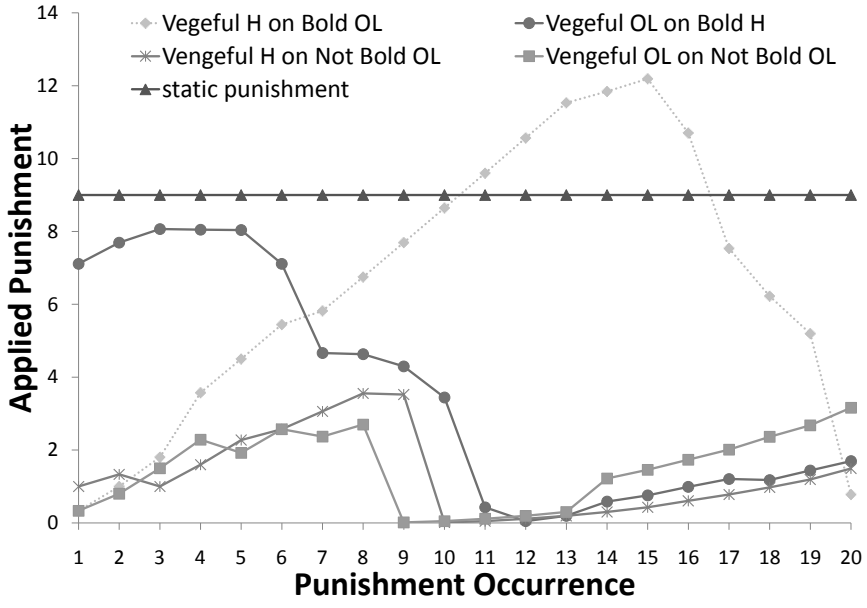


FIGURE 6.5: Punishment value v.s. punishment occurrence: $pu = -1$ (H: Hub; OL: Outlier)

they apply similar punishment values, which makes it look like all of them have come to a punishment agreement on what punishment should be used. This can be seen in-line with some interesting recent thread of research in which agents cooperate with each other to agree on a common punishment value.

6.4 Adaptive Punishment for Limited Observability

Because of the limitations associated with the *realistic* interaction topologies, agents need to be provided with more complex mechanisms than originally introduced to personalise and optimise punishment. Above, we have considered the possibility of *experience-based adaptive punishment*, by which agents are able to determine an appropriate punishment for others based on their prior experience with these others. Experiments with such techniques suggest that it is indeed possible to achieve norm

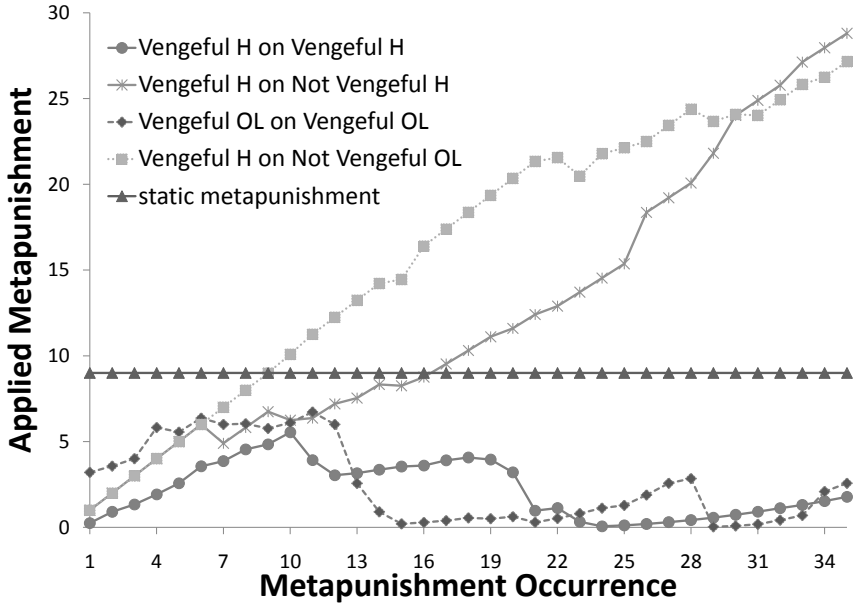


FIGURE 6.6: Metapunishment value v.s. punishment occurrence: $pu = -1$ (H: Hub; OL: Outlier)

establishment with varying levels of punishment, optimising the costs for self-policing. However, in that model, agents interact only with their direct neighbours, building up experience about them through these interactions.

In this section, in contrast, we generalise this so that agents can interact with any other agent in the population, but this is hindered by a lack of experience about these others, since they vary rapidly and are not limited to a small pool. While the consequence of this approach is that experience-based adaptive punishment loses much of its value, the use of *reputation* can offer a solution in this new context. The main focus of this section is thus to consider the use of reputation in supporting adaptive punishment in the context of dyadic interactions.

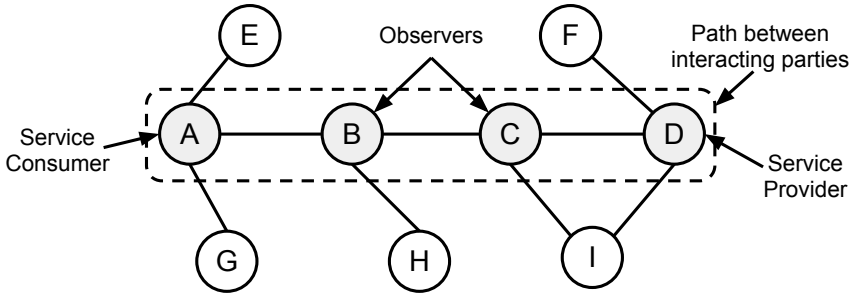


FIGURE 6.7: A one-to-one interaction between A and D

6.4.1 The Metanorm Model and One-to-One Interactions

Our previously introduced models do not capture dyadic interactions that are essential for any P2P-based environment. In order to enable its application to such environments, the model needs to be modified so that it enables emergence of norms when the fundamental principles of interaction are changed. In this section, therefore, we explain how the model is modified in support of this and the results that the new model brings.

6.4.1.1 One-to-One Interactions

Agents in P2P environments must be able to interact with any other agent in the network, yet managing such location and connection between individuals is challenging. When looking for particular services, for example, consumers can ask central registries for information about who can provide them with the service they seek, information that can also be obtained by mobilising neighbours in their social network. Indeed, networks are an effective way to obtain information — for example provided through word-of-mouth — and represent an alternative source, with respect to traditional methods. While conventional approaches in multi-agent systems, such as *registries* or *matchmakers*, partially address this problem [31], in highly dynamic environments, there is a valuable amount of information that cannot be stored in centralised repositories. In

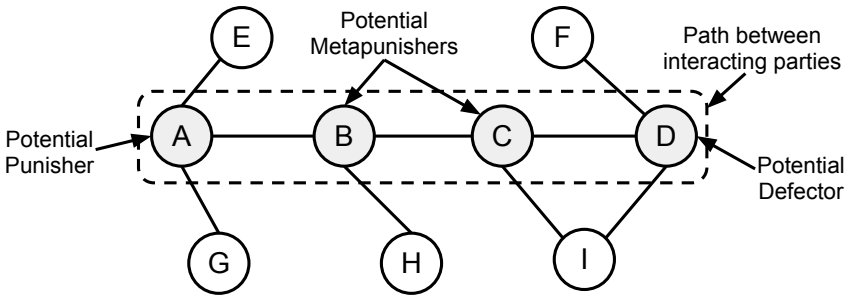


FIGURE 6.8: Potential punishments in a one-to-one interaction between A and D

some cases, much of this information (such as up-to-date details about the quality of service or the availability of the service) may be accessed only by using social networks of interaction. Consequently, we propose a hybrid approach: similarly to the *white pages* of UDDI [26], our agents query a central server to obtain pointers to service providers, and then all other important information about the service is provided by the service providers themselves. The operation of the system is similar to that implemented by Napster.

This can be seen as an agent asking any other agent about a specific file in a peer-to-peer file sharing system. Since the two interacting agents might not be directly connected (they are not neighbours), the interaction takes place through a route that links the two agents but involves various other agents along the way, each of which is capable of observing all communications involved in the interaction. Such a scenario is illustrated in Figure 6.7, which shows an interaction between two agents A and D, where agent A is the service consumer requesting a service from agent D. The dashed line represents the interaction that takes place through a path involving both agents B and C, which are the observers.

With regard to punishment decisions, such changes to the interaction protocol and the observability of interactions affect some decisions, especially in relation to punishment and metapunishment. First, punishment is only applied by the party that is requesting

the service if the service provider defects and does not actually provide the service. This is because the service consumer is the only agent capable of punishing since it is the only agent that is directly affected by the defection. However, other observers record the defection in order to be able to make relevant decisions about subsequent interactions in the future. Second, and as a result of the aim of incentivising agents to respond appropriately to defections, any agent that observes the consumer not punishing the provider for defecting is able to metapunish the consumer. Figure 6.8 shows the example introduced in Figure 6.7, but now in relation to punishment and metapunishment. Since agent D is the service provider for agent A, D is a potential defector (according to its likelihood of defection or *boldness*) towards A. As a result, A is a potential punisher (according to its likelihood of punishing or *vengefulness*) towards D. In addition, and because they are observers of the interaction, both agents B and C are potential metapunishers (again, according to their likelihood of punishing or vengefulness).

6.4.1.2 Experience-Based Adaptive Punishment Results

In light of these modifications, and for the model to reflect the one-to-one interaction scenario, a set of experiments was undertaken to show the effect of this new arrangement on the effectiveness of the model in achieving norm establishment. In these experiments, the population of agents consists of 1,000 agents whose initial boldness and vengefulness are again generated by using a uniform distribution function. With regards to the underlying structure of the system, the focus of our work is on scale-free networks, since it is more representative of the domain of interest (peer-to-peer networks). Agents are thus located in a scale-free network with a starting value of the basic punishment unit being -1. A final remark is that since any communication cost is irrelevant to the phenomenon under investigation, it is assumed that communication required for agents to exchange information is free.

Figure 6.9 illustrates the results obtained from 1,000 independent runs and the same parameter set-up as in Table 6.1, where each point represents the final average boldness and vengefulness of the whole population. This shows that in all runs, the population ends with both high boldness and high vengefulness, which means that agents in this population defect frequently (since they have high boldness) and also punish and metapunish regularly (since they have high vengefulness).

Having analysed the results, it is clear that agents are still defecting, despite the punishments applied, because these punishments do not exceed the utility gain that agents receive from defecting (via the temptation value), and are thus not effective. This is because a punishing agent has insufficient experience with the defecting agent due to limited chances of frequent interaction and the limited memory window size of each agent, preventing the punisher from determining an appropriate punishment to apply to the defector. However, because metapunishment is possible by multiple observers, it guarantees that the level of vengefulness is still high enough for punishment and metapunishment to take place.

This explanation is supported by Figure 6.10, which shows the total punishment applied in each round, and the total temptation gained in the same round. (Note that, for clarity, we have plotted the results of only the first 50 rounds, even though there are actually many more.) The figure shows that the total temptation (utility agents gain from defecting) is significantly higher than the punishment (utility agents lose from defecting), reinforcing our claim that punishment is ineffective.

6.4.2 Reputation-Like Technique

As shown above, in the context of one-to-one interactions, experience-based adaptive punishment alone is not adequate to achieve norm establishment due to the limited

experience of agents with each other. To address this, agents need more information about those involved in the interactions. They can seek such information from other agents that are known to have more experience with the relevant agent (e.g. observers on the interaction path), in a fashion that can be seen as a form of *reputation* for that agent. Agents can then make use of this reputation to determine a more appropriate punishment decision.

Clearly this relies on agents providing truthful reports of the behaviour of others; we could argue that it is in the population's self interest to establish the norm, so that agents are intrinsically motivated to provide reliable information and not lie. However, it is out of the scope of this work to investigate the effects of non-reliable (cheating) agents. In what follows, therefore, we outline a simple mechanism by which an agent establishes reputation for use in determining appropriate punishments. The key point to note is that this is intended not as a sophisticated contribution to work on reputation,

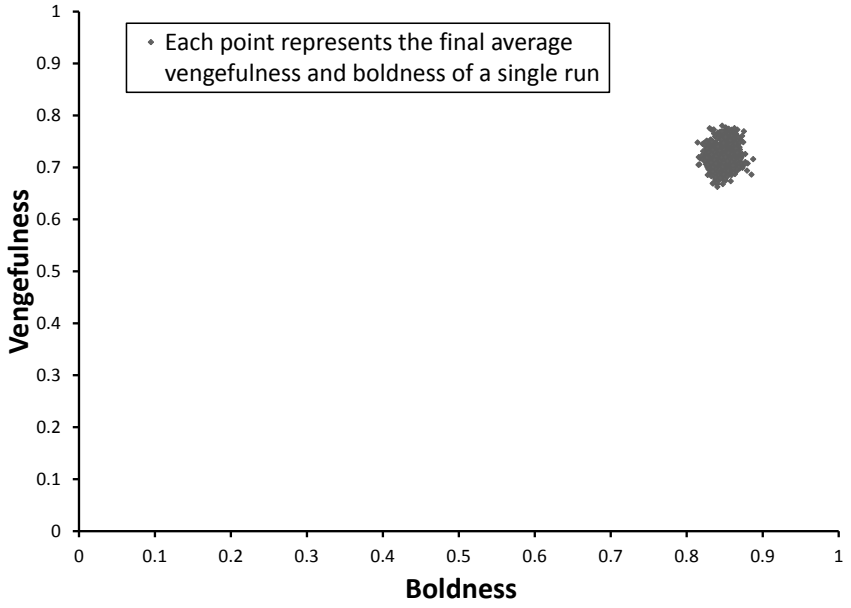


FIGURE 6.9: Adaptive punishment with one-to-one interaction

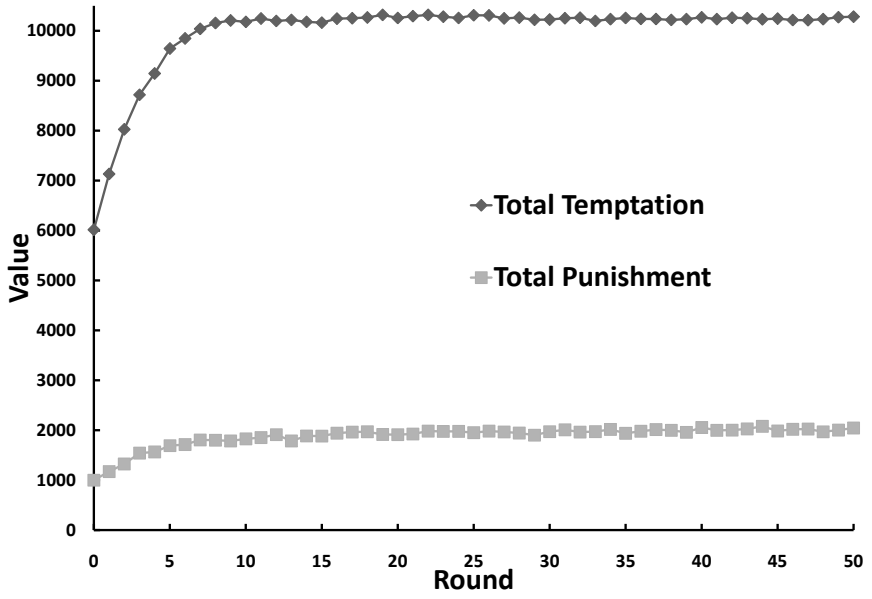


FIGURE 6.10: A Comparison between temptation and punishment levels in each round of a sample run

but as an illustration of how reputation (even in a very simple form) can help to support regulation.

6.4.2.1 Reputation Model

The basic idea of the reputation model is that agents aggregate the information they obtain from others with the information that they already have as a result of their own individual experience. Thus, if agent A decides to punish agent D for defecting, it first sends a request to all agents along the path of interaction asking for information about D, since those agents are likely to have observed various previous interactions of agent D. Then, all other agents (B and C in our example) calculate the defection proportion of D and send it back to A. Having received these different values, A then aggregates them with its own assessment of defection proportion to form a total defection proportion. The aim is to augment the rather limited *local view* of defection

for D with a *global view* that takes into account a broader range of experience to give a more appropriate punishment.

Now, since the method of calculating the local view of defection has already been introduced in Section 6.3.2, here, we introduce the $GlobalView$ that returns the sum of local views of all agents in a path regarding a particular agent, and divide this by the number of agents on the path (to maintain a value between 0 and 1). Formally, $GlobalView$ can be defined as follows:

$$GlobalView : AGENT \times PATH \rightarrow \mathbb{R} \tag{6.1}$$

with:

$$\forall ag_j \in AGENT, \forall p \in PATH : GlobalView(ag_j, p) = \frac{\sum_{ag_k \in p} LocalView(ag_k, ag_j)}{|p|}$$

where:

- $PATH$ is the set of all possible paths (essentially sets of agents along those paths) in the network;
- ag_j is the agent regarding which the local views are aggregated;
- p is the path including the observers of interest; and
- $LocalView(ag_k, ag_j)$ is the local view by agent k of defection of agent j (calculated using the function defined in Section 6.3.2).

The *total* defection view, $TotalView$, can thus be specified by incorporating both the local and global view about the defecting agent, as follows:

$$TotalView : AGENT \times AGENT \times PATH \rightarrow \mathbb{R} \tag{6.2}$$

with: $\forall ag_i, ag_j \in AGENT, \forall p \in IntrPath(ag_i, ag_j)$:

$$TotalView(ag_i, ag_j, p) = \frac{LocalView(ag_i, ag_j) + GlobalView(ag_j, p)}{2}$$

where:

- ag_i is the punishing agent;
- ag_j is the defecting agent; and
- $IntrPath(ag_i, ag_j) \subset PATH$ returns all possible interaction paths between ag_i and ag_j, excluding the interacting parties, i.e.

$$\forall p \in IntrPath(ag_i, ag_j) : ag_i, ag_j \notin p$$

6.4.2.2 Reputation-Based Adaptive Punishment

Given the above, the value of punishment in the reputation-based punishment technique can be calculated similarly to experience-based punishment, but with the replacement of the local defection view with the total defection view, as follows:

$$RepExpPunish : AGENT \times AGENT \times PATH \rightarrow \mathbb{R} \tag{6.3}$$

with: $\forall ag_i, ag_j \in AGENT, \forall p \in IntrPath(ag_i, ag_j)$:

$$RepExpPunish(ag_i, ag_j, p) = TotalView(ag_i, ag_j, p) \times pu$$

where ag_i is the punishing agent, ag_j is the defecting agent, and p is the interaction path of interest.

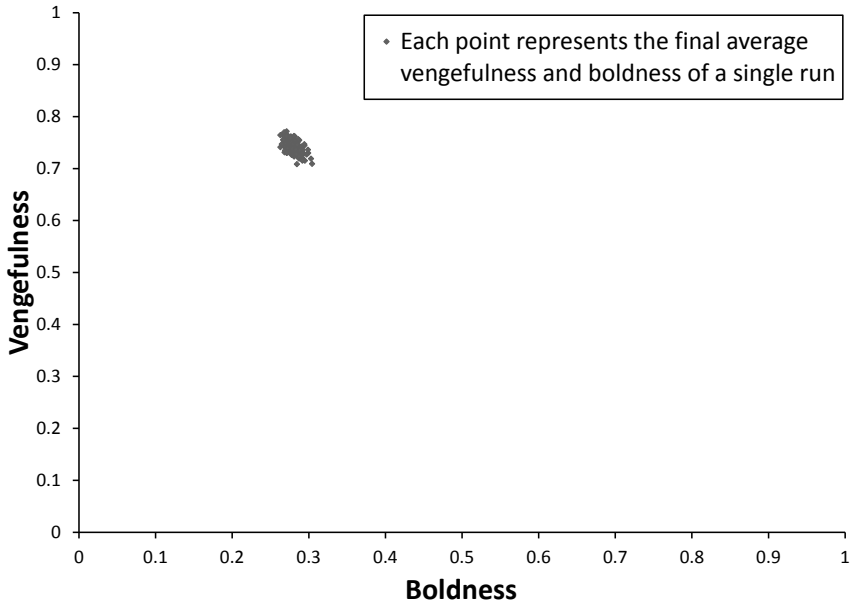


FIGURE 6.11: Reputation-based punishment with one-to-one interaction

6.4.3 Results

Given this new reputation-based model, we ran experiments to show the effect of the technique on the defection rate. These experiments were set up like those introduced earlier (using the parameters in Table 6.1), and provided results indicating the successful application of levels of punishment appropriate to establishing the norm. A sample result of one experiment involving 1,000 runs is shown in Figure 6.11, in which there is a noticeable improvement in the results. First, the population still has a high level of vengefulness, which means that punishment and metapunishment are active. Second, and most importantly, the level of boldness has dropped significantly, because agents that are faced with a defecting agent can gather much more information about this agent. As a result, their punishment can be much more appropriate in limiting the opportunities for this agent to defect in the future.

As before, Figure 6.12 shows the total temptation and punishment in each round.

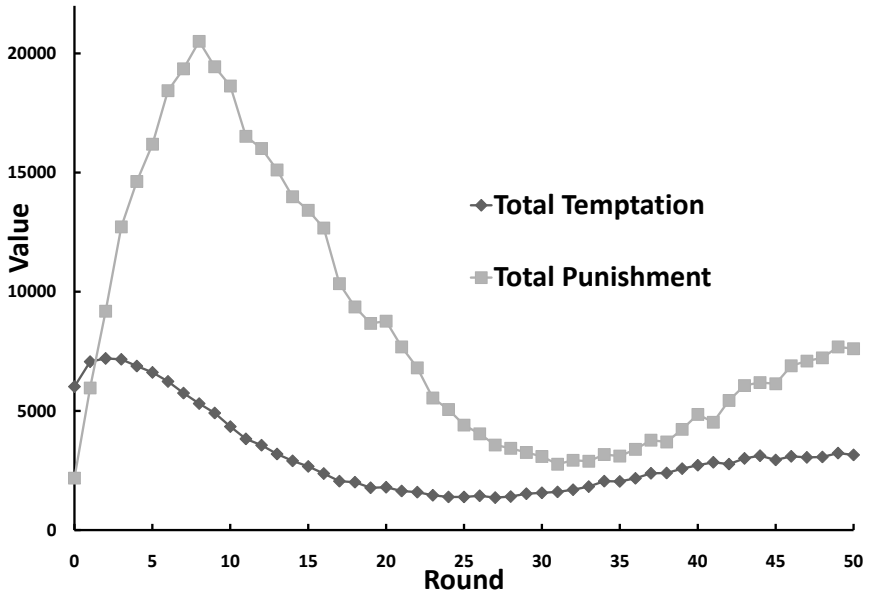


FIGURE 6.12: A comparison between temptation and punishment levels in each round of a sample run

This time, however, the results indicate that punishment does indeed exceed temptation, suggesting that punishment is much more appropriate in preventing agents from defecting. Interestingly, the punishment values vary significantly over time, also indicating that they are continually adjusted in line with experience to give the level most suitable for the circumstances.

6.5 Conclusion and Future Work

Norm emergence is an important and valuable phenomenon that has applications to self-organising systems such as peer-to-peer networks and wireless sensor networks in which there is no interference from any central or outside authority. While there has been much work on this phenomenon (as discussed earlier), punishment has generally been considered to be static (though with some exceptions). With such static

approaches, determining an appropriate level of punishment is difficult, with punishments being excessive in some cases. Importantly, such punishments can themselves decrease the utility of the overall system, by reducing the number of participants (as in P2P networks, for example), bringing counterproductive effects. In response, we described a mechanism for determining punishment values dynamically, using the prior experience of agents with those they are interacting, in order to specify the appropriate level. Through simulation experiments and results, we have shown that our technique helps to achieve norm emergence, but at a much lower cost to the system in terms of the punishment incurred, bringing benefits to the society or the system as a whole, and improving efficiency in ways that suggest potential value in real-world cases.

Furthermore, we investigated the effect of such adaptive punishment on establishing the norm in the context of limited observability. In particular, our results show that experience-based adaptive punishment fails to stop agents defecting when observation is limited. This is due to agents not having enough information about each other, so they are not able to estimate efficient punishments. However, by introducing reputation into the model to provide extra information, this does not remain the case. Reputation enriches agents' information, and allows them to determine appropriate punishment decisions that are able to regulate agent behaviour and prevent them from defecting. While our reputation model is very simple, it is perfectly adequate for our aim of investigating the use of reputation in building an adaptive punishment mechanism. Having seen that reputation can be valuable in the development of such a mechanism, our future work will focus on investigating the use of more complex reputation models [108, 125, 147, 154, 155] and their effect on improving the efficiency of adaptive punishment even further.

Chapter 7

Case Study: P2P File Sharing System

7.1 Introduction

While we have demonstrated that the models developed over the previous chapters are effective in different environments (and topologies), we have not situated these models in any specific real-world context. In this chapter, therefore, we seek to demonstrate the performance of these models, in just such a setting, that of peer-to-peer (P2P) file sharing, as exemplified by *Gnutella* (introduced in Chapter 1, so we do not repeat an extensive description here).

In *Gnutella*, peers are both clients and servers that request and provide files from and to others. If a peer does not have a file, it can pass a request to others in turn, and when a requested file is found, it is returned down the path of requesting peers. There is no payment involved and no limit on the number or proportion of files that can be accessed or shared. However, sharing files consumes bandwidth so peers may choose not

to do so, while still accessing the files of others. This *free riding* problem is prevalent in such P2P networks. In this chapter, we apply the previous models in this scenario to better evaluate them. In what follows, we first the P2P file sharing scenario detailing the integration of our models within it. This is followed with evaluation showing the effect that the models have on such scenario, before the chapter is finally concluded.

7.2 The P2P File Sharing Scenario

7.2.1 Assumptions

We begin by introducing some key assumptions on which our case study is based. First, when a peer wants to access a particular file, it needs to locate another peer that already has that file. Various techniques can be used to achieve this, but since this is not the focus of our work, we assume the existence of a local registry that monitors a subset of peers in some part of the network and is able to communicate with other local registries nearby. Each registry keeps track of the files that are obtained by each peer within its subset so that it is able to report back to a requesting peer which of its known peers possess the requested file.

While we consider all the previously developed models here, the use of reputation in the model of Chapter 6 is based on the assumption of accurate reputation information. In general, accurate reputation is achieved only by sophisticated reputation techniques (see [59, 154, 155] for examples), but since this is not the focus of our work, in this case study peers blindly trust information provided to them by others. (One way to address this could be to integrate the FairTorrent protocol [62], which allows nodes to exchange information about other peers in a trustworthy way. Much more could be written about these different techniques, but that would be a diversion away from the purpose of this chapter, so we say no more about it here.)

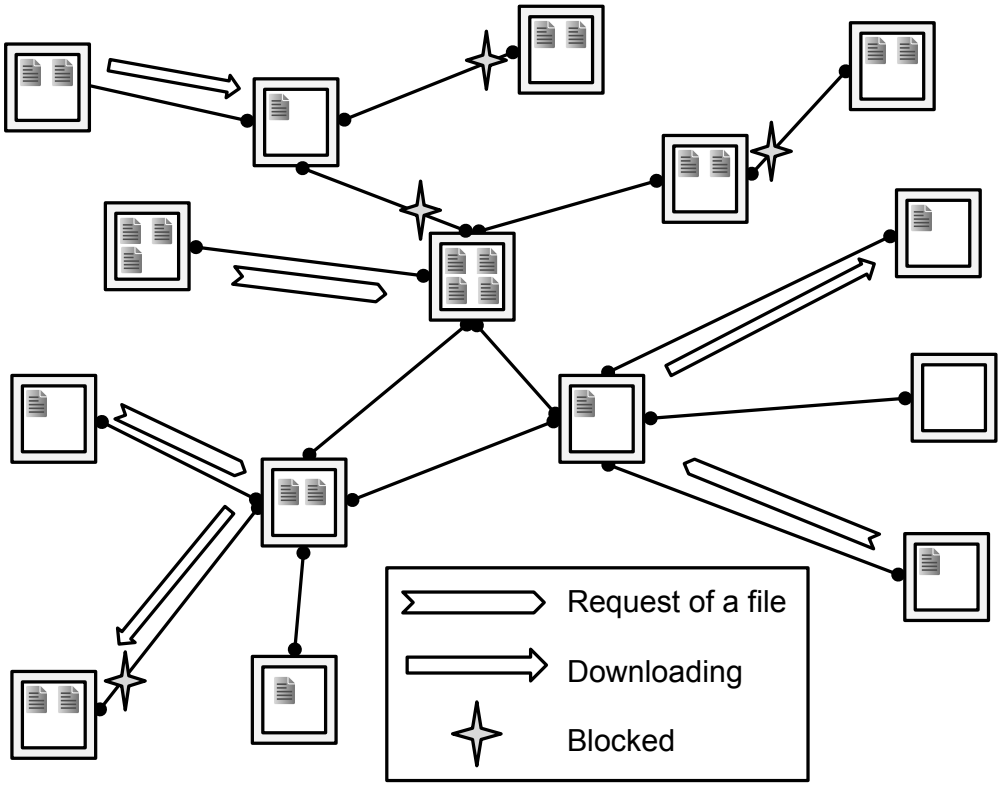


FIGURE 7.1: The P2P File Sharing Scenario

7.2.2 Scenario Parameters

Given these assumptions, illustrated in Figure 7.1, we can summarise the scenario as follows. Several agents are involved in a P2P file sharing community, with the connections between them forming a scale free network [86] (discussed in Chapter 5). These agents share certain files, with each agent being initialised with a subset of these files according to some content popularity distribution; popularity is generally highly skewed, with some files having a higher popularity (more demand for them) than others, thus with multiple copies in the population of agents. This is intuitive when you compare high demand content (e.g. the latest film releases) against low demand content (e.g. music from the 1950s). As such, a *long-tail* emerges in which many items are only requested once or twice (or never!).

Interactions between agents are repeated over multiple rounds, in each of which, requests are generated at a rate determined by a Poisson Distribution and assigned to agents using a uniform distribution function. An agent makes a request based on the content popularity distribution, and then locates all other agents in the community that have the relevant file, using a central registry. It then sends file requests to a specified maximum number of agents, where each request is a separate thread between the requesting agent and its target. This takes place through the shortest path between the two agents, in which all other agents on the path play the role of *observer*, introduced to encourage cooperation in the community.

In our simulations, the number of agents in the population is fixed at some value, $NumA$, the number of files shared is fixed at $NumF$ with each agent's subset numbering $NumFA$, according to the popularity distribution, PD. The request rate is represented as λ, and the maximum number of agents to which requests can be sent is μ.

7.2.3 Model Behaviour

As we have seen, an agent that receives a request decides, according to its cooperation strategy (boldness B), whether to share the file with the requesting agent. If it does share the file, then the requesting agent receives the file (in one round for simplicity) and the central repository is updated to reflect this. If not then, according to its punishment strategy (vengefulness V), the requesting agent chooses whether to punish (block or reduce its download rate) the defecting agent with a certain punishment P bearing in mind that a punishing agent pays a specific enforcement cost E. If the requesting agent does not punish, then it risks being metapunished by all observers, who also record the decision for future consideration. At the end of each round, agents reconsider their policies according to the scores obtained and update their policies accordingly.

TABLE 7.1: Parameter Initialisation

Parameter	**Description**	**Value**
$NumA$	Number of agents	1000
$NumF$	Number of files	$40,000$
PD	Probability distribution of files	Zipf-Mandelbrot (fetch-at-most-once) with skew power $\alpha = 1$
λ	No. of requests per round per agent	2
μ	No. of peers to request file from	3
B	Probability of rejecting request	Uniform distribution from 0 to 1
V	Probability of punishing a peer	Uniform distribution from 0 to 1
T	Temptation utility that can be gained from defecting	3

As specified within the model in Chapter 7, adaptive punishment is used by agents based on the information available, which mainly comes from their prior experience and from the reputation obtained from the *observers* of the interaction. Thus, in this scenario, an agent a that seeks to punish another agent b for not providing a file, uses information from its memory about b's past responses to a's requests, as well as information from the observers on the interaction path to form a reputation value, and constructs an appropriate punishment decision.

7.3 Evaluation

While we aim to show the effect of using our models within a complex setting, we also want to show the effect of the various methods of punishment that have been investigated in the thesis. In consequence, the results presented here involve, in addition to reputation-based adaptive punishment, those obtained from applying both static punishment and memory-based adaptive punishment. First, however, we provide the parameter set-up used in all experiments.

7.3.1 Parameter Set-up

Table 7.1 specifies the values assigned to all parameters used in all instances of our simulation. The parameters and their assignments are not our own invention; instead, in order to provide a sound basis on which to consider our work, these have been adopted from the work of Gummadi et al. [53] on evaluating P2P systems. In addition, in our experiments, we used these values, but varied the values of punishment to use -1 and -9 in order to show the differences resulting from these extremes.

Now, in order to ensure the results are meaningful in the context of the case study, we introduce the following terms. First, norm emergence here means that peers should be always responding positively to requests to share files received from others. According to our analysis, this is only the case when peers have a low tendency to reject requests (low boldness) and a high tendency to punish those that do not (high vengefulness). This is referred to as *file sharing establishment* in the results introduced later. Conversely, when peers have a high tendency to reject any incoming requests and a low tendency to spend resources on punishing those that do the same, the situation is referred to as *free riding establishment.*

7.3.2 Overall Results

We undertook experiments with the punishment techniques identified above, the results of which are shown in the graph of Figure 7.2. Each symbol in the graph represents the average of the results of 1000 runs of each technique. From the graph, we can clearly see that both memory-based punishment with a punishment unit of -1 and static punishment with a punishment unit of -1 result in free riding establishment. This is due to the punishments applied being insufficient to persuade defecting agents to share files. In the case of static punishment, a punishment unit of -1 is insufficient

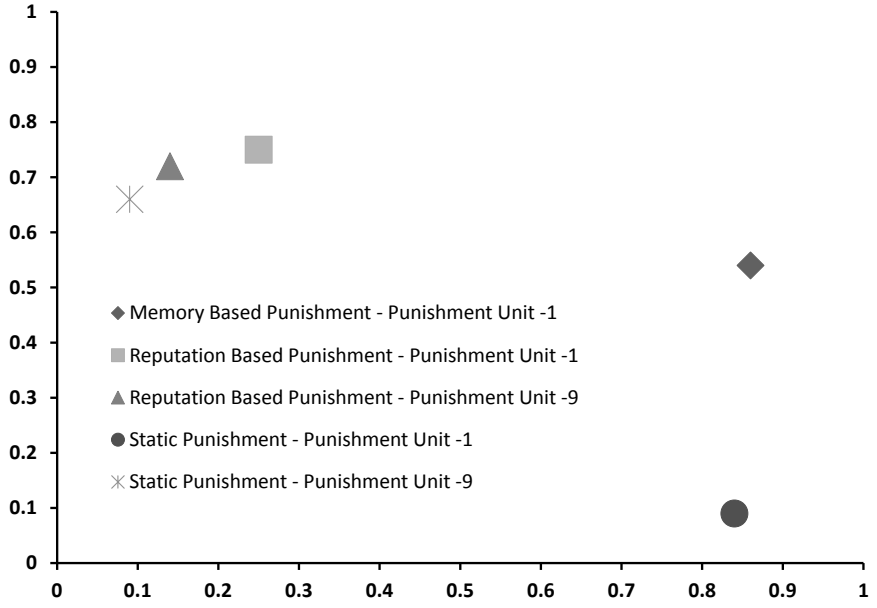


FIGURE 7.2: Overall Results - Each point represents the average of 1000 runs

to overcome the utility gained from defecting, which is 3 in this case, so peers tend to defect. Moreover, this reasoning also applies to the tendency to punish others, since the static metapunishment cost of -1 is also insufficient to balance the loss inured by the enforcement cost of -2.

With regard to memory-based punishment, the initial punishment unit of -1 is insufficient to overcome the utility gained from defecting. Here, punishment should increase with the experience that agents gain through their interactions with each other but, as discussed in Chapter 6, the lack of repeated interactions (a well known phenomenon in such domains [63]), prevents this. With respect to punishing defecting peers, memory-based punishment results in midrange vengefulness in comparison to the static technique. This is because there is a reasonable chance that agents will be in the path of the interaction, as observers, of the same agent multiple times within the current memory window, and there are usually multiple observers in each interaction. This allows agents to increase the metapunishment appropriately to overcome the enforcement cost

in some instances. However, this is only enough to keep vengefulness in the midrange area.

In terms of static punishment with a punishment unit of -9, the norm can be considered established, since all runs ended with file sharing establishment. This is simply because the punishment applied is sufficient to convince agents to sacrifice some of their bandwidth to upload files in order to avoid the more extreme punishment that can be incurred otherwise, especially with the threat of metapunishment encouraging punishment of free riders. Similarly, reputation-based punishment with punishment units of -1 and -9 both succeed in establishing file sharing in the population. This is because in both cases, reputation helps to overcome the lack of repeated interactions.

Before providing more detail of the nature of the punishments involved in these simulations, and their effects, we first provide details of the effectiveness of the techniques, in terms of the proportion of file requests accepted.

7.3.3 File Requests Acceptance Rate

We have seen that reputation-based punishment with both high and low punishment units, and static punishment with a high punishment unit, succeed in eliminating free riders, while static punishment and memory based punishment both with low punishment units fail. Here, we examine the cause a little more, but considering the evolution of the way in which peers respond to file requests. Figure 7.3 shows the percentage of file requests that are accepted in the first 1000 timesteps of a sample run from each of the different techniques whose overall results are shown above.

From the figure, it can be easily seen that in both the static punishment with a low punishment unit and memory-based punishment with a low punishment unit, the acceptance rate is very low across the whole run (represented through the first 1000

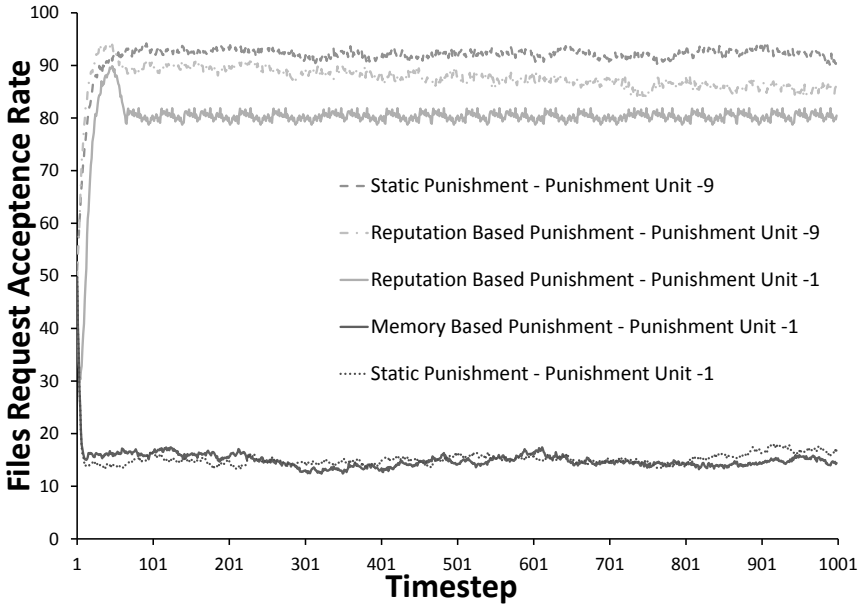


FIGURE 7.3: File Request Acceptance Rate

timsteps shown in the figures), giving rise to the free riding establishment. Conversely, reputation-based adaptive punishment with both punishment units manages to achieve an acceptance rate that is very close to the static punishment with a high punishment unit; when the punishment unit is high, this is much closer than when it is low.

7.3.4 Individual Punishments and Metapunishments

We can also consider the changes that arise with respect to punishment, but here we focus only on a comparison between static punishment with a high punishment unit and reputation-based adaptive punishment with a low punishment unit, because these succeed in establishing the norm, and the latter covers the same case with a high punishment unit. In what follows, we distinguish between punishment applied to peers that refuse to share files, which we consider in turn.

7.3.4.1 Individual punishment

Figure 7.4 illustrates the changes to punishment applied to a peer a that has a high tendency to reject requests, and also provides a comparison between adaptive punishment with a punishment unit of -1 and static punishment with a punishment unit of -9. From this figure, it can be observed that until around 40 time steps, punishment initially increases but is subject to dramatic fluctuation within this overall trend when using the adaptive punishment approach. This is due to two main reasons. First, at the start of the run there is insufficient reputation information about peer a, which can only be constructed from repeated interaction, repeated appearance of a in the path of interactions, and repeated defection by a. Second, the chance of an observer with such reputation information about a defector being on the path of interaction increases through the run.

In addition, it can be observed that after sufficient reputation has been established for peer a, punishment increases to a maximum level of 22, which is more than double the punishment that is used in the static punishment approach. This does not last long, and drops to a level close to zero once agent a responds appropriately by not defecting. However, at a later point in time, punishment starts to increase again, at timestep 154, almost 115 timsteps later. (Note that we only show a punishment occurrences (when punishment is greater than zero) in the figure, which explains gaps such as between 39 and 154, where no punishment took a place). This is simply because the natural exploration in the model leads to a sudden increase in boldness and to defect once more. However, at this point, a has had time to repair its reputation by sharing files, resulting in punishment starting from a low level again. This time, punishment increases more slowly because instead of starting from a position of no information, it now starts from a position of having a good reputation through its immediately prior history of compliance.

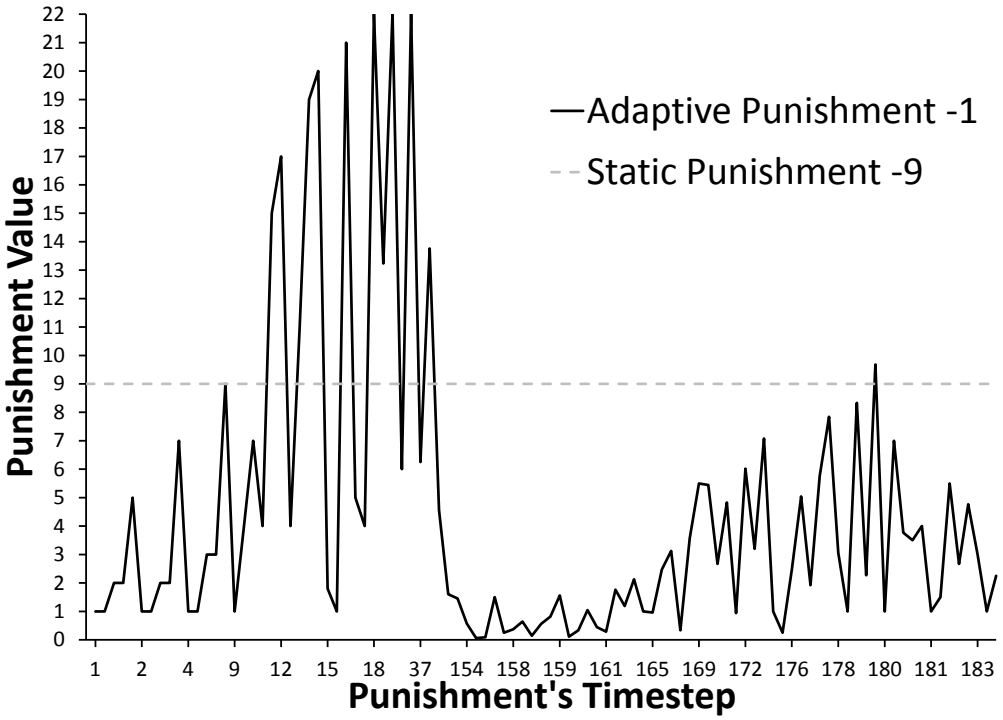


FIGURE 7.4: Individual Punishment on High Boldness Agent

7.3.4.2 Individual Metapunishments

Finally, we consider how metapunishment affects the tendency of peers to punish others through analysing the change in metapunishment applied to a peer b, which has a low tendency to punish others, as shown in Figure 7.5. In this case, metapunishment starts to increase after the few timesteps that are required for peer b to establish a poor reputation for not punishing peers that do not share files with others. Such reputation is determined by the number of interactions in which b has requested files, the number of times in which it has not punished those that did not share files it needed, the number of times that b has previously been observed, and the number of observers doing so.

Similarly to the case above, it can be realised that metapunishment decreases when agent b starts to respond. However, it increases again 52 timesteps later, at about 74,

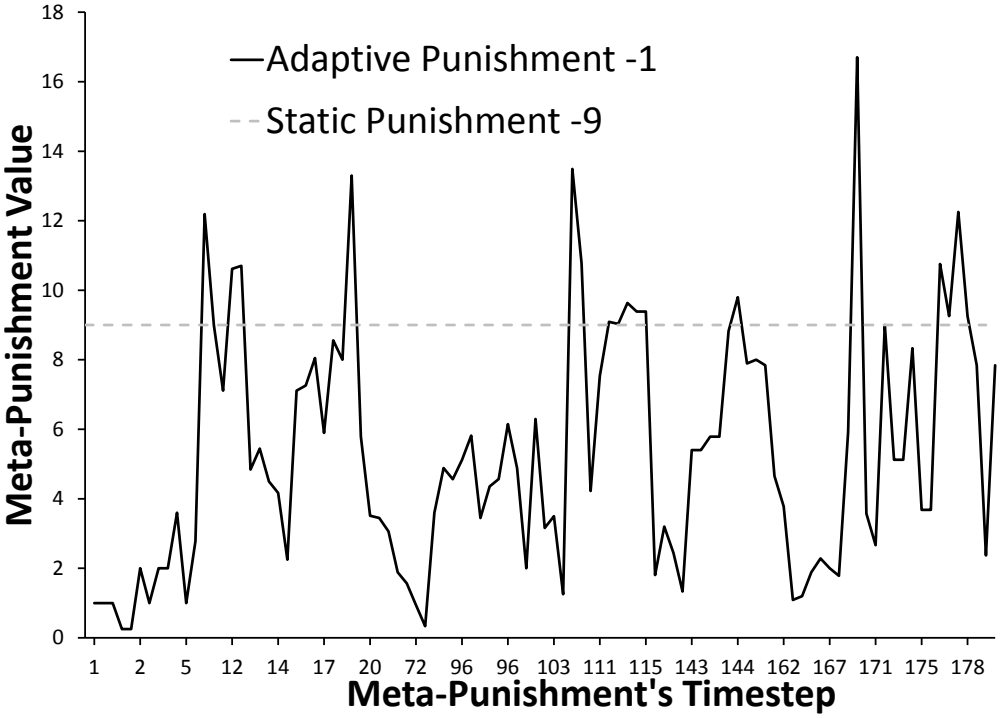


FIGURE 7.5: Individual Metapunishment on low Vengefulness Agent

with no metapunishment within. Again, it takes longer for metapunishment to increase to high level due to the positive reputation that agent b has gained from punishing other agents that do not share files. In total, it can be observed that the used metapunishment is generally less than in the static punishment case, which supports that adaptive punishment is capable of achieving same results with less overall metapunishment.

7.4 Conclusion

The free-riding phenomenon is a well recognised problem in P2P file sharing networks, since it allows members of a network to damage it by accessing available files over the network without sharing them with others, threatening the availability of these files and the performance of the network as a whole. In this respect, free-riding provides a

valuable opportunity to examine the capability of the models developed in this thesis. In order to provide an effective study, integrating our models in just such a P2P file sharing system, several domain-specific aspects have required consideration, such as the existence of files, methods of distributing files among agents, and mechanisms for generating requests for files. We have reported on these aspects for completeness and to provide confidence in the integrity of our results. In this context, various experiments were undertaken and their results analysed.

The results themselves are valuable. When using metanorms with a suitable punishment unit, with either static or adaptive punishment, the results show that a file sharing norm is established in the population and, in addition, that the percentage of accepted file requests increases to almost 90%, reflecting a greater tendency for agents to share their files with others. In term of efficiency, the results show that adaptive punishment with a low punishment unit is capable of achieving the same level of cooperation among agents as static punishment with a high punishment unit, but that overall, much less punishment is used in the former case. These results simply demonstrate that even when our models are placed in (simulated) real world scenarios, they are still capable of achieving cooperation among self-interested entities with great efficiency.

Chapter 8

Conclusion and Future Work

8.1 Introduction

Regulating the behaviour of self-interested agents is a well recognised problem in the field of agents and multi-agent systems. Many researchers have investigated various approaches to achieve such a purpose. Norms have been suggested by many as a valuable mechanism to turn agents' attention away from acting maliciously. Some work has introduced an authority that is capable of designing new norms and imposing them on the agent population, which is known as the top-down approach. However, such an approach is not always possible especially in vast and dynamic environments in which no one central authority is computationally capable of monitoring the behaviour of a huge population. As a result, others have suggested to use the bottom-up approach in which norms are allowed to emerge through agents' interactions, and the agents themselves monitor their behaviour and apply punishments to each other to isolate undesired behaviour, which results in some sort of united behaviour emerging in the population; this is known as norm emergence.

Norm emergence has also recently been recognised by many as an efficient approach for establishing cooperation among self-interested agents. In particular, individual punishment has been recognised as a very effective method to bring about norm emergence. However, there are limitations in the work presented in the literature. In particular, the use of metanorms has almost been ignored despite the great potential that it shows. In addition, work on examining the effect of specific structures that constrain agents' interactions with each other has received little attention. Finally, work on the use of punishment as a method for achieving norm emergence has focused on examining the effect of static punishment and has totally ignored the possibility of integrating an adaptive punishment technique. These issues are those on which this thesis has focussed, and they are outlined in more detail in the following summary.

8.2 Summary

8.2.1 Norm Emergence and Axelrod

As we have seen, the use of *altruistic* punishment as a means to encourage the emergence of norms in a population of self interested agents was first proposed by Axelrod. This involves agents punishing each other for their malicious behaviour, while incurring enforcement costs for doing so. Axelrod was able to show that punishment alone is insufficient to establish norms, due to the lack of motivation for agents to employ it, especially because of the enforcement costs incurred. This led to the proposal to use *metanorms*, which enable agents to punish those that are observed not to punish defecting agents. Axelrod's experiments showed that with the help of such metanorms, norm establishment becomes possible. However, his model was limited in adopting various assumptions — in many cases, implicit and unstated assumptions — that severely limit its application to computational systems; his experiments were also overly simplistic, so

that while his results point to the potential of the approach, they do not convince that the obtained results are significant for the development of real computational systems.

While some researchers (e.g. [42, 103]) have tried to investigate the use of metanorms in more complex scenarios (and in particular Galan et al. [42], who show that the use of metanorms does not produce valuable results in the long term), their work merely raises questions that demand further investigation and clarification. In this respect, and in support of the need to develop mechanisms to support norm establishment in computational systems, we initially undertook our own analysis of Axelrod's model, through the reimplementation described in Chapter 3, to investigate model and its weaknesses. Similar to Galan et al., our results showed that Axelrod's model does not succeed in achieving norm emergence in the long term as a result of the *noise* generated from the evolutionary approach adopted, magnified by the use of *mutation*. Importantly, our consideration of the model revealed several distinct weaknesses that were addressed in the subsequent chapters of this thesis.

8.2.2 Centralised and Decentralised Emergence

Fundamental to Axelrod's model is the requirement for a central authority that has full control over the entire agent population, manipulating it by adding and removing agents. However, in computational systems in which agents represent the interests of different people or organisations, where no single authority is capable of exercising such control, this is inappropriate. To address this, in Chapter 4 we introduced an alternative to Axelrod, maintaining the key aspects, but removing this centralisation by replacing the central evolutionary approach with a distributed individual learning approach in which the agents themselves modify their behaviour. This was achieved through the use of the novel techniques of *strategy copying* and *reinforcement learning*.

Two kinds of strategy copying were investigated: copying from the best agent in terms of the obtained score (which gives promising results in the short term but fails in the long term); and copying a random strategy from a group of agents considered to perform well (giving good results). This latter technique gives rise to norm emergence in both the short term and long term, but suffers from two drawbacks. First, it assumes access to the private strategy of other agents, which is unreasonable both because it may be constrained by a system design, and because in general this is not accessible with self-interested agents. Second, it produces norm collapse when introducing the constraint that any agent seeking to metapunish another for not punishing a defecting agent must first observe the defection of the first agent.

In response, in Chapter 4 we also specified a reinforcement learning (RL) algorithm that allows each agent to adapt its own strategy without the need to access those of other agents. Our evaluation showed that it manages to achieve norm emergence with this observation constraint, enabling us to move from a model that gives rise to occasional norm collapse to one in which norm emergence is guaranteed, as illustrated in Figure 8.1.

8.2.3 Topological Structure

While addressing the problems arising from the fundamental nature of Axelrod's model, we also sought to address its limitations arising from the unrealistic assumption that all agents are fully connected so that they can all interact and observe every other agent in the population. Such an assumption is not possible in general for computational systems for many reasons. First, the number of agents in such systems can be so large that it is impossible to manage the traffic generated from their interactions nor to observe all other agents. Second, the connections between agents are subject to different network topologies not considered by Axelrod. To address these issues, Chapter 5

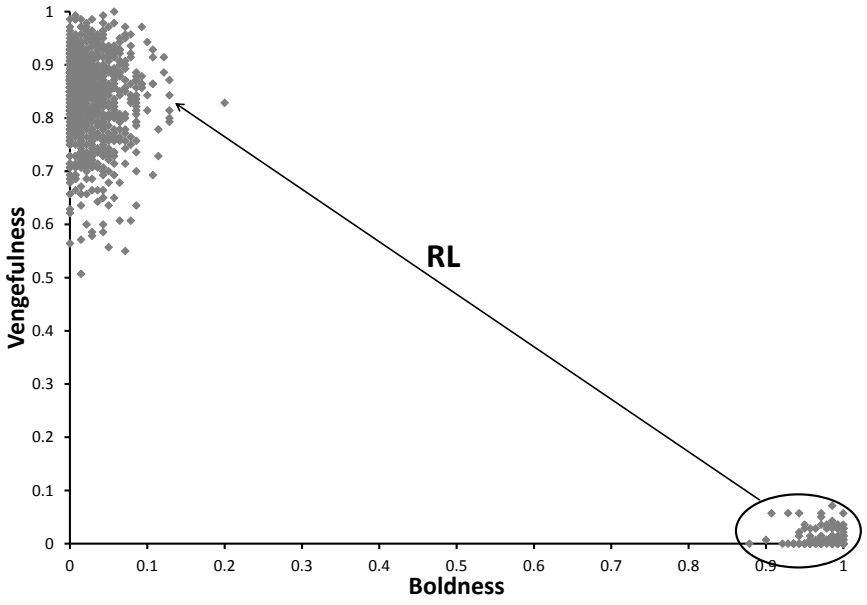


FIGURE 8.1: Evolutionary to Reinforcement Learning

aimed at adapting the model to be effective with different interaction topologies and their effects on norm emergence. We were able to show our model from Chapter 4 achieves norm emergence over both lattices and small world networks, but not in scale free networks, due to the nature of the connection distribution between hub and outlier nodes. Due to the vast number of connections of hubs in scale free networks, their interactions are much more frequent than outliers, causing them to be the only agents that learn as a result.

In consequence, we further refined our model by removing the restrictions on learning which had been constrained to apply only to poorly performing agents, and avoiding access to the performance of others, which is unreasonable in computational systems. Instead, in Chapter 5 we introduced a new *universal learning* (UL) technique that allows all agents to improve performance, but this produced only norm collapse, due to the constraints imposed by the limited observability of outlier behaviour. In fact, this

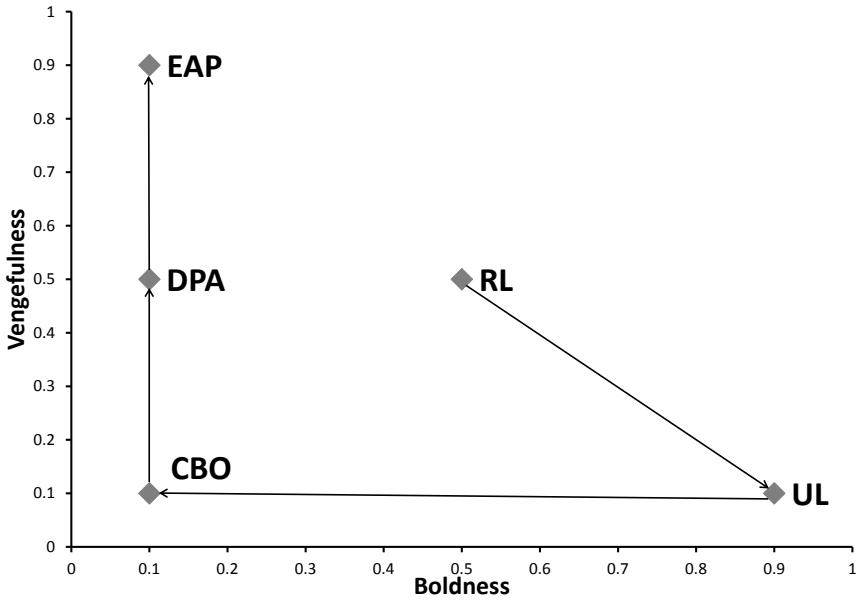


FIGURE 8.2: Scale Free Results' Enhancement

revealed another inadequacy with Axelrod's original formulation of a uniform probability of being seen. In real systems, observability is restricted by network connections rather than some arbitrary probability distribution; to address this we adopted a *connection-based observation* (CBO) technique that also had the consequence of reducing the tendency of agents to defect, while maintaining their tendency to punish defectors.

Despite this, the problems arising from scale free networks remained, due to the their asymmetric nature. Using an approach in which a uniform degree of modification to strategy was thus ineffective, so we refined our techniques once more to adjust the amount of learning in relation to performance through *dynamic policy adaptation* (DPA), bringing about the desired behaviour and norm emergence.

8.2.4 Adaptive Punishment

Finally, in Chapter 6, we considered the efficiency of the application of punishment in seeking to avoid excessive constraint on system behaviour, so that agents can change the amount of punishment according to the case at hand. Thus, agents that defect rarely should not incur the same punishment as those that defect regularly. This led to the *experiential adaptive punishment* (EAP) technique by which agents track their experience with other agents and determine punishments accordingly, enabling a much more effective use of metapunishment, and bringing near optimal convergence to norm establishment.

Now, in seeking to ensure that this is applicable to, for example, peer-to-peer file sharing systems, we introduced some further considerations, imposed by such systems. In particular, since repeated interactions between the same pair of agents is unlikely, potentially bringing about norm collapse, we introduced *reputation-based adaptive punishment* (RAP) as a means of providing an agent with the information required to determine how much to punish another in a single interaction, and with the punishing agent being constrained by the threat of metapunishment from those observers providing it with this reputation information.

The progression through the various different models across all these chapters is summarised and illustrated in Figure 8.2. The figure clearly shows the performance of the models and their associated approaches in moving from poor performance to the near optimal performance of EAP. In addition, Figure 8.3 shows how this is adjusted further with RAP in the case of systems, like P2P, in which repeat interactions are unlikely, moving from normal collapse to norm establishment.

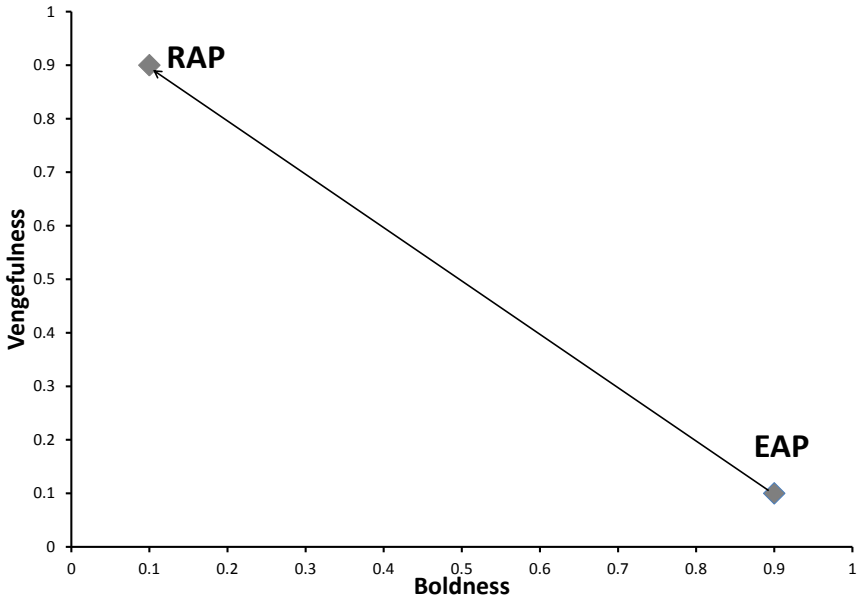


FIGURE 8.3: Adaptive Punishment and Dyadic Interactions

8.3 Contributions

Given our summary above, in this section we enumerate the specific contributions made by the work reported in this thesis.

- We have provided a model that is capable of achieving norm emergence, eliminating the need for a central control authority that requires access to private information and manipulates the population accordingly. This is achieved through a reinforcement learning algorithm that allows each agent to adapt its own policies based on available local information.

- We have provided a model that integrates the concept of network topologies, a critical component of real computational systems, extending the work of Axelrod so that it is effective in such environments. This model has been demonstrated to be effective with lattices, small worlds and scale free networks.

- We have provided a novel adaptive punishment technique that adjusts the level of punishment according to need, avoiding excessive constraint on system behaviour that could jeopardise the effectiveness of the overall system, while ensuring that norm emergence can still be achieved.

In addition, and in consequence there are some secondary contributions that also merit explicit mention.

- We have revealed and overcome major weaknesses in the Axelrod's influential metanorm model, preventing its direct use in encouraging cooperative behaviour in real computational systems. These weaknesses include omniscient agents, failure in the long term due to side effects of the adopted evolutionary approach, the assumption of fully connected networks, and the use of exaggerated static punishment.
- We have eliminated the problems arising from overloaded hubs in scale free networks by allowing agents to adapt their behaviour appropriately in light of information obtained from previous interactions.
- We have provided a solution to the problem of lack of experience in systems with random dyadic interactions (like P2P), through the use of reputation-based adaptive punishment.

8.4 Limitations

Clearly, given limited time, and limited focus, there are inevitably several aspects of our work that are limited to some extent. We summarise these below.

8.4.1 Simple Reputation Model

In Chapter 6, the reputation-based adaptive punishment technique adopts a specific reputation model. Although the experimental results show that this manages to achieve norm establishment, it is limited in that it blindly trusts information provided by others to build the desired reputation value. An agent that wants to punish another, requests from all the agents observing the interaction the information they have about the agent under punishment. Then, the requesting agent accumulates the received information with its own information in order to build the overall reputation. Someone may argue that the requesting agent can assign weights to the other agents, representing its degree of trust in their provided information. These weights can then be used to build the overall reputation. This is an interesting view that we aim to adopt in future work. However, since the focus of the work here is to demonstrate that reputation can indeed support adaptive punishment decisions, a simple reputation model is sufficient. However, we do agree that much more complex reputation models would make the model applicable in more realistic settings.

8.4.2 Identity Change and Whitewashing

The models developed in this thesis assume that agents have static identities that do not change over time. However, this ignores the phenomenon of identity change and *whitewashing* by which agents leave a system, and rejoin with a new identity in order to improve their reputation. This is a common problem, in particular in the P2P arena. While we have not addressed this since it is not the focus of our work, various techniques have been suggested to overcome this problem [23, 38], and these could be integrated into our models.

8.5 Future Work

In addition to limitations, there are also other directions that could be explored in the future. We have identified various paths that can be taken in moving this work further forward.

8.5.1 Dynamic Temptation

While we have considered dynamic punishments, we ignored some other aspects that might also be dynamic. For example, we might also seek to make temptation dynamic in the sense that its value might vary according to circumstance. This could make sense in the context of P2P file sharing, where the saving from defection varies depending on the size of the file that is not shared (not sharing a 1GB file provides a greater temptation than not sharing a 1MB file). Introducing such aspects in our models would make it even more realistic in terms of simulating real-world behaviour.

8.5.2 Richer Reputation Models

As mentioned in Chapter 6 and Section 8.4.1, our reputation model is rather simplistic, but serves our purpose of analysing its value in support of norm establishment. However, more sophisticated reputation models [59, 154, 155] might consider other aspects, including questions of whom to ask for information, how much trust should be placed in the information received from different sources, and how to aggregate such information and build the punishment decision.

8.5.3 Reward Schemes

In the models developed through this thesis, we have focussed on punishment in line with Axelrod's model. However, an alternative incentive mechanism to encourage co-operation is the use of reward schemes in which agents are rewarded for cooperative behaviour, rather than penalised for defecting. An analysis of such schemes would be an interesting possibility to investigate in the future.

8.5.4 Reputation as a Punishment

The use of reputation itself as a form of punishment or reward is also an interesting possibility. Thus, instead of purely using material punishment by which agents incur costs, an agent's reputation could be increased or decreased as a result of cooperative or malicious behaviour. Such an approach might be used on its own or in support of material punishment. In the former case, punishment would be related only to altering reputation, while in the latter case, altering reputation is used as a starting point before using material punishment, or after it is found that the material punishment does not work.

8.5.5 Dynamic Networks

Having analysed various network structures and their effect in Chapter 5, all the networks considered are static, where connections between agents do not change, and agents maintain the same set of connections through the whole simulation. However, this is not always the case, since there are many scenarios in which connections are dynamic. In some cases, such dynamism is due to the nature of the system. For example, in systems where mobile agents exist, agent connections are frequently changing according to agent locations. In other cases, changes might be a mechanism that agents

use to encourage cooperation or even exploit the system itself. For example, an agent might change one of its connections, if this connection is linking it to a malicious agent, which can be seen as an extreme type of punishment that might lead to complete isolation of the malicious agent. Conversely, a malicious agent might decide to entirely change its location in a network either as a result of establishing a bad reputation in its local population, or because of the desire of exploiting another part of the network. Analysing the performance of the metanorm model over such dynamic structures is important to support the model in terms of its scalability and applicability in various scenarios.

References

[1] E. Adar and B. A. Huberman. Free riding on gnutella. *First Monday*, 5(10), 2000.

[2] G. Andrighetto, M. Campenni, R. Conte, and M. Paolucci. On the Immergence of Norms: a Normative Agent Architecture. In *Proceedings of AAAI Symposium, Social and Organizational Aspects of Intelligence*, 2007.

[3] G. Andrighetto and D. Villatoro. Beyond the carrot and stick approach to enforcement: An agent-based model. In B. Kokinov, A. Karmiloff-Smith, and N. J. Nersessian, editors, *European Conference on Cognitive Science*, 2011.

[4] G. Antoniou, D. Billington, G. Governatori, and M.J. Maher. A flexible framework for defeasible logics. In *PROCEEDINGS of the American National Conference on Artificial Intelligence*, pages 405–410, 2000.

[5] G. Antoniou, D. Billington, G. Governatori, and M.J. Maher. Representation results for defeasible logic. *ACM Transactions on Computational Logic*, 2(2):255–287, 2001.

[6] M. S. Artigas. Disconnection punishment in trust bootstrapping: Benefits of activity stereotypes. In *P2P 2012: Proceedings of the 12th International Conference on Peer-to-Peer Computing*, pages 149–154. IEEE, 2012.

[7] R. Axelrod. An evolutionary approach to norms. *The American Political Science Review*, 80(4):1095–1111, 1986.

[8] R. M. Axelrod. *The evolution of cooperation.* Basic Books, 1984.

[9] A. L. Barabasi and R. Albert. Emergence of Scaling in Random Networks. *Science*, 286(5439):509–512, 1999.

[10] D. Billington. Defeasible logic is stable. *Journal of Logic and Computation*, 3(4):379–400, 1993.

[11] A. Blanc, Y. Liu, and A. Vahdat. Designing incentives for peer-to-peer routing. In *INFOCOM 2005: Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 1, pages 374–385, 2005.

[12] G. Boella and L. van der Torre. Permissions and obligations in hierarchical normative systems. In *ICAIL 2003: Proceedings of the 9th International Conference on Artificial Intelligence and Law*, pages 109–118, 2003.

[13] G. Boella and L. van der Torre. Fulfilling or Violating Obligations in Normative Multiagent Systems. In *IAT 2004: Proceedings of IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pages 483–486. IEEE Computer Society, 2004.

[14] G. Boella and L. van der Torre. Substantive and procedural norms in normative multiagent systems. *Journals of Applied Logic*, 6(2):152–171, 2008.

[15] G. Boella, L. van der Torre, and H. Verhagen. Introduction to normative multiagent systems. *Computational & Mathematical Organization Theory*, 12(2-3):71–79, 2006.

[16] M. Boman. Norms as constraints on real-time autonomous agent action. In *Multi-Agent Rationality: Proceedings of the Eighth European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, volume 1237 of *Lecture Notes in Computer Science*, pages 36–44. Springer, 1997.

[17] M. Boman. Norms in artificial decision making. *Artificial Intelligence and Law*, 7(1):17–35, 1999.

[18] E. Borenstein and E. Ruppin. Enhancing autonomous agents evolution with learning by imitation. *Interdisciplinary Journal of Artificial Intelligence and the Simulation of Behavior*, 1(4):335–348, 2003.

[19] M. Bowling and M. Veloso. Rational and convergent learning in stochastic games. In *IJCAI 2001: Proceedings of the 17th International Joint Conference on Artificial Intelligence*, pages 1021–1026, 2001.

[20] J.M. Broersen, F.P.M. Dignum, M.V. Dignum, and J-J.Ch. Meyer. Designing a deontic logic of deadlines. In *DEON 2004: Proceedings of the 7th International Workshop on Deontic Logic in Computer Science*, volume 3065, pages 43–56. Lecture Notes in Computer Science, 2004.

[21] J. Carmo and A. Jones. A new approach to contrary-to-duty obligations. *Defeasible Deontic Logic*, 263:317–344, 1997.

[22] J. Carpenter, P. Matthews, and O. Ong'ong'a. Why punish? social reciprocity and the enforcement of prosocial norms. *Journal of Evolutionary Economics*, 14(14):407—-429, 2004.

[23] J. Chen, H. Lu, and S. D. Bruda. A solution for whitewashing in P2P systems based on observation preorder. In *Proceedings of the 2009 International Conference on Networks Security, Wireless Communications and Trusted Computing - Volume 02*, NSWCTC '09, pages 547–550. IEEE Computer Society, 2009.

[24] R. Conte and C. Castelfranchi. Understanding the functions of norms in social groups through simulation. *Artificial Societies The Computer Simulation of Social Life*, pages 252–267, 1995.

[25] R. Conte, C. Castelfranchi, and F. Dignum. Autonomous Norm Acceptance. In *ATAL '98: Proceedings of the 5th International Workshop on Intelligent Agents V, Agent Theories, Architectures, and Languages*, pages 99–112, 1999.

[26] F. Curbera, M. Duftler, R. Khalaf, W. Nagy, N. Mukhi, and S. Weerawarana. Unraveling the web services web: An introduction to soap, wsdl, and uddi. *IEEE Internet Computing*, 6:86–93, 2002.

[27] K. Dautenhahn and C. L. Nehaniv, editors. *Imitation in animals and artifacts.* MIT Press, Cambridge, MA, USA, 2002.

[28] J. Davis, P. Laughlin, and S. Komorita. The Social Psychology of Small Groups: Cooperative and Mixed-Motive Interaction. *Annual Review of Psychology*, 27(1):501–541, 1976.

[29] A. P. de Pinninck, C. Sierra, and M. Schorlemmer. Distributed Norm Enforcement: Ostracism in Open MultiAgent Systems. *Computable Models of the Law: Languages, Dialogues, Games, Ontologies*, pages 275–290, 2008.

[30] A. P. de Pinninck, C. Sierra, and W. M. Schorlemmer. Friends no more: norm enforcement in multiagent systems. In *Proceedings of the Sixth International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 640–642, 2007.

[31] K. Decker, K. P. Sycara, and M. Williamson. Middle-agents for the internet. In *IJCAI 1997: Proceedings of the 15th International Joint Conference on Artificial Intelligence*, pages 578–583, 1997.

[32] J. Delgado. Emergence of social conventions in complex networks. *Artificial Intelligence*, 141(1-2):171–185, October 2002.

[33] J. Delgado, J. M. Pujol, and R. Sangüesa. Emergence of coordination in scale-free networks. *Web Intelligence and Agent Systems*, 1:131–138, 2003.

[34] F. Dignum. Autonomous agents with norms. *Artificial Intelligence and Law*, 7(1):69–79, 1999.

[35] F. Dignum and D. Kinny. From desires, obligations and norms to goals. *Cognitive Science Quarterly*, 2(3-4):407—-427, 2002.

[36] J. M. Epstein. Learning to be thoughtless: Social norms and individual computation. *Computational Economics*, 18(1):9–24, 2001.

[37] E. Fehr and S. Gachter. Altruistic punishment in humans. *Nature*, 415:137–140, 2002.

[38] M. Feldman, C. Papadimitriou, J. Chuang, and I. Stoica. Free-riding and whitewashing in peer-to-peer systems. In *PINS 2004: Proceedings of the ACM SIGCOMM workshop on Practice and Theory of Incentives in Networked Systems*, pages 228–236, New York, NY, USA, 2004. ACM.

[39] J. Fix, C. von Scheve, and D. Moldt. Emotion-based norm enforcement and maintenance in multi-agent systems: foundations and petri net modeling. In *AAMAS '06: Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 105–107. ACM, 2006.

[40] F. Flentge, D. Polani, and T. Uthmann. Modelling the emergence of possession norms using memes. *Journal of Artificial Societies and Social Simulation*, 4(4), 2001.

[41] H. Franks, N. Griffiths, and A. Jhumka. Manipulating convention emergence using influencer agents. *Autonomous Agents and Multi-Agent Systems*, pages 1–39, 2012.

[42] J. M. Galan and L. R. Izquierdo. Appearances can be deceiving: Lessons learned re-implementing Axelrod's evolutionary approach to norms. *Journal of Artificial Societies and Social Simulation*, 8(3), 2005.

[43] F. Giardini, G. Andrighetto, and R. Conte. A cognitive model of punishment. In S. Ohlsson & R. Catrambone, editor, *Proceedings of the 32nd Annual Conference of the Cognitive Science Society Austin, TX: Cognitive Science Society*, pages 1282–1288. Portland, Oregon, 2010.

[44] J.P. Gibbs. Norms: The problem of definition and classification. *The American Journal of Sociology*, 70(5):586–594, 1965.

[45] M. C. Gonz'alez, P. G. Lind, and H. J. Herrmannc. Networks based on collisions among mobile agents. *Physica D-nonlinear Phenomena*, 224:137–148, 2006.

[46] G. Governatori and A. Rotolo. Defeasible Logic: Agency, Intention and Obligation. In *Proceedings of the Seventh International Workshop on Deontic Logic in Computer Science*, pages 114–128, 2004.

[47] G. Governatori and A. Rotolo. BIO Logical Agents: Norms, Beliefs, Intentions in Defeasible Logic. *Autonomous Agents and Multi-agent Systems*, 17(1):36–69, 2008.

[48] A. Grizard, L. Vercouter, T. Stratulat, and G. Muller. *A Peer-to-Peer Normative System to Achieve Social Order*, pages 274–289. Springer-Verlag, Berlin, Heidelberg, 2007.

[49] D. Grossi, H. Aldewereld, and F. Dignum. Ubi lex, ibi poena: Designing norm enforcement in e-institutions. In *In Coordination, Organizations, Institutions, and Norms in Multi-Agent Systems II*, pages 107–120. Springer, 2006.

[50] D. Grossi, H. Aldewereld, J. Vázquez-Salceda, and F. Dignum. Ontological aspects of the implementation of norms in agent-based electronic institutions. *Computational and Mathematical Organization Theory*, 12(25):251–275, 2006.

[51] D. Grossi, H. M. Aldewereld, and F. Dignum. Ubi lex, ibi poena: Designing norm enforcement in e-institutions. In P. Noriega, J. Vázquez-Salceda, G. Boella, O. Boissier, M.V. Dignum, N. Fornara, and E. Matson, editors, *Coordination, Organizations, Institutions, and Norms in Agent Systems II*, pages 101–114. Springer, 2007.

[52] R. Guerraoui, K. Huguenin, A. Kermarrec, and M. Monod. On Tracking Freeriders in Gossip Protocols. In *P2P 2009: Proceedings of the 9th International Conference on Peer-to-Peer Computing*, 2009.

[53] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, and J. Zahorjan. Measurement, modeling, and analysis of a peer-to-peer file-sharing workload. In *SOSP 2003: Proceedings of the 19th ACM symposium on Operating Systems Principles*, pages 314–329, 2003.

[54] G. Hayes and J. Demiris. A robot controller using learning by imitation. In *Proceedings of the 2nd International Symposium on Intelligent Robotic Systems*, pages 198–204, 1994.

[55] G. He and J. C. Hou. Tracking targets with quality in wireless sensor networks. In *ICNP '05: Proceedings of the 13TH IEEE International Conference on Network Protocols*, pages 63–74. IEEE Computer Society, 2005.

[56] D. Helbing, A. Szolnoki, M. Perc, and G. Szab. Punish, but not too hard: how costly punishment spreads in the spatial public goods game. *New Journal of Physics*, 12(8):083005, 2010.

[57] J. M. Helmhout, H. W. M. Gazendam, and R. J. Jorna. Control over emergence. In *AISB 2008: Proceedings of the Convention: Communication, Interaction, and Social Intelligence*, pages 1—-8, 2008.

[58] J. Henrich and R. Boyd. Why people punish defectors. weak conformist transmission can stabilize costly enforcement of norms in cooperative dilemmas. *Journal of Theoretical Biology*, 208):79–89, 2001.

[59] Z. Hu, H. Lin, and Y. Zhou. A fuzzy reputation management system with punishment mechanism for P2P network. *Journal of Networks*, 6(2):190–197, 2011.

[60] K. Jaffe and L. Zaballa. Co-operative punishment cements social cohesion. *Journal of Artificial Societies and Social Simulation*, 13:3, 2010.

[61] M. Kandori, G. J. Mailath, and R. Rob. Learning, mutation, and long run equilibria in games. *Econometrica*, 61(1):29–56, 1993.

[62] S. Kaune. *Performance and Availability in Peer-to-Peer Content Distribution Systems: A Case for a Multilateral Incentive Approach.* PhD thesis, TU Darmstadt, March 2011.

[63] S. Kaune, G. Tyson, K. Pussep, A. U. Mauthe, and R. Steinmetz. The seeder promotion problem: Measurements, analysis and solution space. In *ICCCN 2010: Proceedings of the 19th International Conference on Computer Communications and Networks*, pages 1–8. IEEE, 2010.

[64] J. Kittock. Emergent conventions and the structure of multi–agent systems. In *Lectures in Complex systems: the proceedings of the 1993 Complex Systems*

Summer School, Santa Fe Institute Studies in the Sciences of Complexity Lecture Volume VI, Santa Fe Institute, pages 507–521. Addison-Wesley, 1995.

[65] M. J. Kollingbaum and T. J. Norman. Norm adoption in the NoA agent architecture. In *AAMAS '03: Proceedings of the Second International Joint Conference on Autonomous Agents and MultiAgent Systems*, pages 1038–1039. ACM, 2003.

[66] X. Y. Koushik, K. Niyogi, S. Mehrotra, and N. Venkatasubramanian. Adaptive target tracking in sensor networks. In *CNDS '04: Communication Networks and Distributed Systems Modeling and Simulation Conference*, 2004.

[67] R. Krishnan, D. M. Smith, Z. Tang, and R. Telang. The impact of free-riding on peer-to-peer networks. In *HICSS '04: Proceedings of the 37th Annual Hawaii International Conference on System Sciences*, page 70199.3. IEEE Computer Society, 2004.

[68] K. Kulakowski. The norm game: punishing enemies and not friends. *Journal of Economic Interaction and Coordination*, 4(1):27–37, 2009.

[69] K. Kulakowski and P. Gawronski. To cooperate or to defect? altruism and reputation. *Physica A: Statistical Mechanics and its Applications*, 388(17):3581–3584, 2009.

[70] K. Lakkaraju and L. Gasser. Norm emergence in complex ambiguous situations. In *COIN 2008: Proceedings of the AAAI Workshop on Coordination, Organizations, Institutions, and Norms*, 2008.

[71] D.G. Lawrence. Procedural norms and tolerance: A reassessment. *The American Political Science Review*, 70:80—-100, 1976.

[72] G. Lokhorst. Mally's deontic logic. *Stanford Encyclopedia of Philosophy*, 2004.

[73] F. Lopez y Lopez, M. Luck, and M. d'Inverno. Normative agent reasoning in dynamic societies. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '04, pages 732–739, Washington, DC, USA, 2004. IEEE Computer Society.

[74] S. Mahmoud, N. Griffiths, J. Keppens, and M. Luck. An analysis of norm emergence in Axelrod's model. In *NorMAS'10: Proceedings of the Fifth International Workshop on Normative Multi-Agent Systems*. AISB, 2010.

[75] S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. An analysis of norm emergence in axelrod's model. In *EUMAS'10: Proceedings of the 8th European Workshop on Multi-Agent Systems*, 2010.

[76] S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Norm establishment via metanorms in network topologies. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, volume 3, pages 25–28, 2011.

[77] S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Overcoming omniscience in Axelrod's model. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, volume 3, pages 29–32, 2011.

[78] S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Efficient norm emergence through experiential dynamic punishment. In *Proceedings of the 20th European Conference on Artificial Intelligence*, pages 576–581. IOS Press, 2012.

[79] S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Establishing norms for network topologies. In B. van Riemsdijk S. Cranefield, J. Vazquez-Salceda and P. Noriega, editors, *Coordination, Organizations, Institutions and Norms in Agent Systems IV*, volume 7254 of *Lecture Notes in Computer Science*, 2012.

[80] S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Overcoming hub effects in scale free networks. In *Proceedings of the Fourteenth International Workshop on Coordination, Organisations, Institutions and Norms (COIN 2012)*, pages 136–150, 2012.

[81] S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Overcoming omniscience for norm emergence in Axelrod's metanorm model. In B. van Riemsdijk S. Cranefield, J. Vazquez-Salceda and P. Noriega, editors, *Coordination, Organizations, Institutions and Norms in Agent Systems IV*, volume 7254 of *Lecture Notes in Computer Science*, 2012.

[82] S. Mahmoud, J. Keppens, N. Griffiths, and M. Luck. Norm emergence through dynamic policy adaptation in scale free networks. In *Coordination, Organizations, Institutions and Norms in Agent Systems V*, 2013. to appear.

[83] S. Mahmoud, D. Villatoro, J. Keppens, and M. Luck. Optimised reputation-based adaptive punishment for limited observability. In *Sixth IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, 2012.

[84] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson. Wireless sensor networks for habitat monitoring. In *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 88–97. ACM, 2002.

[85] D. Makinson and L. van der Torre. What is Input/Output Logic? Input/Output Logic, Constraints, Permissions. In *Normative Multi-agent Systems*. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), 2007.

[86] R. Matei, A. Iamnitchi, and P. Foster. Mapping the gnutella network. *Internet Computing, IEEE*, 6(1):50 –57, 2002.

[87] P. McNamara. Deontic logic. *Stanford Encyclopedia of Philosophy*, 2006. http://plato.stanford.edu/archives/spr2006/entries/logic-deontic.

[88] P. Meara. Emergent properties of multilingual lexicons. *APPLIED LINGUISTICS*, 27(4):620–644, 2006.

[89] A. Milenković, C. Otto, and E. Jovanov. Wireless sensor networks for personal health monitoring: Issues and an implementation. *Computer Communications*, 29(13-14):2521–2533, 2006.

[90] P. Mukherjee, S. Sen, and S. Airiau. Emergence of norms with biased interactions in heterogeneous agent societies. In *Web Intelligence and Intelligent Agent Technology Workshops, 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, pages 512–515, 2007.

[91] D. Mungovan, E. Howley, and J. Duggan. The influence of random interactions and decision heuristics on norm evolution in social networks. *Computational & Mathematical Organization Theory*, 17(2):152–178, 2011.

[92] M. Nakamaru and U. Dieckmann. Runaway selection for cooperation and strict-and-severe punishment. *Journal of theoretical biology*, 257(1):1–8, 2009.

[93] M. E. J. Newman. The Structure and Function of Complex Networks. *SIAM Review*, 45(2):167–256, 2003.

[94] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113, 2004.

[95] N Nikiforakis. Punishment and counter-punishment in public good games: Can we really govern ourselves? *Journal of Public Economics*, 92:91–112, 2008.

[96] D. Nute. Defeasible logic. In *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 3. Oxford University Press, 1987.

[97] D. Nute, editor. *Defeasible Deontic Logic*. Kluwer, Dordrecht, 1997.

[98] F. Y. Okuyama, R. H. Bordini, and A. C. da Rocha Costa. Spatially distributed normative objects. In *Proceedings of Environments for Multi-Agent Systems III, Third International Workshop*, 2007.

[99] C. O'Riordan, A. Cunningham, and H. Sorensen. Emergence of cooperation in n-player games on small world networks. In S. Bullock, J. Noble, R. Watson, and M. A. Bedau, editors, *Artificial Life XI: Proceedings of the Eleventh International Conference on the Simulation and Synthesis of Living Systems*, pages 436–442. MIT Press, Cambridge, MA, 2008.

[100] C. O'Riordan and H. Sorensen. Stable cooperation in the n-player prisoner's dilemma: The importance of community structure. In Karl Tuyls, Ann Nowe, Zahia Guessoum, and Daniel Kudenko, editors, *Adaptive Agents and Multi-Agent Systems III. Adaptation and Multi-Agent Learning*, volume 4865 of *Lecture Notes in Computer Science*, pages 157–168. Springer Berlin Heidelberg, 2008.

[101] R. Posner and E. Rasmusen. Creating and enforcing norms, with special reference to sanctions. Law and Economics 9907004, EconWPA, Jul 1999.

[102] H. Prakken and M. Sergot. Contrary-to-duty obligations. *Studia Logica*, 57:91–115, 1996.

[103] M. J. Prietula and D. Conway. The evolution of metanorms: quis custodiet ipsos custodes? *Computational and Mathematical Organization Theory*, 15(3):147–168, 2009.

[104] K. Pripužić, H. Belani, and M. Vuković. Early forest fire detection with sensor networks: Sliding window skylines approach. In *KES 2008: Proceedings of the 12th International Conference on Knowledge-Based Intelligent Information and Engineering Systems, Part I*, pages 725–732. Springer-Verlag, 2008.

[105] L. Ramaswamy and L. Liu. Free riding: A new challenge to peer-to-peer file sharing systems. In *HICSS '03: Proceedings of the 36th Annual Hawaii International Conference on System Sciences*, pages 220–220. Springer, 2003.

[106] R. Riolo, M. Cohen, and R. Axelrod. Evolution of cooperation without reciprocity. *Nature*, 414:441–443, 2001.

[107] Young U. Ryu and Ronald M. Lee. Defeasible deontic reasoning: A logic programming model. In *Deontic Logic in Computer Science*, 1993.

[108] J. Sabater and C. Sierra. REGRET: reputation in gregarious societies. In *AGENTS '01: Proceedings of the Fifth International Conference on Autonomous Agents*, pages 194–195. ACM, 2001.

[109] N. Salazar, J. A. Rodriguez-Aguilar, and J. L. Arcos. Robust coordination in large convention spaces. *AI Commun.*, 23:357–372, December 2010.

[110] B. T. R. Savarimuthu, S. Cranefield, M. Purvis, and M. Purvis. Norm emergence in agent societies formed by dynamically changing networks. In *IAT '07: Proceedings of the 2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pages 464–470, 2007.

[111] B. T. R. Savarimuthu, S. Cranefield, M. Purvis, and M. Purvis. Role model based mechanism for norm emergence in artificial agent societies. In *COIN '07: Proceedings of the International Workshop on Coordination, Organization, Institutions and Norms*, pages 1–12, 2007.

[112] B. T. R. Savarimuthu, Stephen Cranefield, Maryam Purvis, and Martin Purvis. Role model based mechanism for norm emergence in artificial agent societies. In *Coordination, Organizations, Institutions, and Norms in Agent Systems III, COIN 2007 International Workshops*, volume 4870 of *Lecture Notes in Computer Science*, pages 203–217. Springer, 2008.

[113] B. T. R. Savarimuthu, M. Purvis, M. Purvis, and S. Cranefield. Social norm emergence in virtual agent societies. In M. Baldoni, T. C. Son, M. B. van Riemsdijk, and M. Winikoff, editors, *Declarative Agent Languages and Technologies VI*, volume 5397 of *Lecture Notes in Computer Science*, pages 18–28. Springer, 2009.

[114] R. Schollmeier. A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications. In *P2P 2001: Proceedings of the 1st International Conference on Peer-to-Peer Computing*, pages 101–102. IEEE Computer Society, 2001.

[115] O. Sen and S. Sen. Effects of social network topology and options on norm emergence. In *Proceedings of the Fifth International Conference on Coordination, Organizations, Institutions, and Norms in Agent Systems*, pages 211–222, 2010.

[116] S. Sen and S. Airiau. Emergence of norms through social learning. *IJCAI 2007: Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 1507–1512, 2007.

[117] S. Sen and S. Airiau. Emergence of norms through social learning. In *IJCAI 2007: Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 1507–1512. Morgan Kaufmann Publishers Inc., 2007.

[118] Y. Shoham and M. Tennenholtz. Emergent Conventions in Multi-Agent Systems: Initial Experimental Results and Observations (Preliminary Report). In *Proceedings of the 3rd International Conference on KR&R*, pages 225–232, 1992.

[119] Y. Shoham and M. Tennenholtz. Co-Learning and the Evolution of Social Activity. Technical report, Stanford, 1993.

[120] Y. Shoham and M. Tennenholtz. On social laws for artificial agent societies: off-line design. *Artificial Intelligence*, 73(1-2):231–252, February 1995.

[121] Y. Shoham and M. Tennenholtz. On the emergence of social conventions: modeling, analysis, and simulations. *Artificial Intelligence*, 94:139–166, 1997.

[122] T. Slembeck. Reputations and fairness in bargaining - experimental evidence from a repeated ultimatum game with fixed opponents. Experimental, EconWPA, May 1999.

[123] M. Song, R. Chen, and j. An. Social conventions to promise learning convergence. In *FSKD '07: Proceedings of the Fourth International Conference on Fuzzy Systems and Knowledge Discovery*, pages 660–662, Washington, DC, USA, 2007. IEEE Computer Society.

[124] Luc Steels. The origins of ontologies and communication conventions in multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 1:169–194, 1997.

[125] W. T. L. Teacy, J. Patel, N. R. Jennings, and M. Luck. TRAVOS: Trust and reputation in the context of inaccurate information sources. *Journal of Autonomous Agents and Multi-Agent Systems*, 12:2006, 2006.

[126] G. Therborn. Back to norms! on the scope and dynamics of norms and normative action. *Current Sociology*, 50:863–880, 2002.

[127] Richmond H. Thomason. Deontic Logic as Founded on Tense Logic. In Risto Hilpinen, editor, *New studies in deontic logic: norms, actions, and the foundations of ethics*, pages 165–176. D. Reidel Publishing Company, Dordrecht, Holland, 1981.

[128] J. E. Tomberlin. Contrary-to-duty imperatives and conditional obligation. *Noûs*, 15(3):357–375, 1981.

[129] Christian Traxler and Joachim Winter. Survey evidence on conditional norm enforcement. Technical Report 3, Max Planck Institute for Research on Collective Goods, 2009.

[130] R. Tuomela. *The Importance of Us: A Philosophical Study of Basic Social Norms.* Stanford University Press, Stanford, CA, United States of America, 1995.

[131] R. Tuomela and M. Bonnevier-Tuomela. Norms and agreement. *European Journal of Law, Philosophy and Computer Science*, 41–46(5), 1995.

[132] E. Ullman-Margalit. *The Emergence of Norms.* Clarendon Press, Oxford, 1977.

[133] Paulo Urbano, João Balsa, Luis Antunes, and Luís Moniz. Force versus majority: A comparison in convention emergence efficiency. In *Coordination, Organizations, Institutions and Norms in Agent Systems IV: COIN 2008 International Workshops, COIN@AAMAS 2008, Estoril, Portugal, May 12, 2008. COIN@AAAI 2008, Chicago, USA, July 14, 2008. Revised Selected Papers*, pages 48–63, 2008.

[134] V. Urovi, S. Bromuri, K. Stathis, and A. Artikis. Initial steps towards run-time support for norm-governed systems. In *COIN@AAMAS'10: Proceedings of the 6th International Workshop on Coordination, Organizations, Institutions, and Norms in Agent Systems*, pages 268–284, 2011.

[135] L. van der Torre. Contextual Deontic Logic: Normative Agents, Violations and Independence. *Annals of Mathematics and Artificial Intelligence*, 37(1):33–63, 2003.

[136] L. van der Torre and Y. Tan. The many faces of defeasibility in defeasible deontic logic. *Defeasible Deontic Logic*, pages 79–121, 1997.

[137] L. van der Torre and Y. Tan. Contrary-to-duty reasoning with preference-based dyadic obligations. *Annals of Mathematics and Artificial Intelligence*, 27(1–4):49–78, 1999.

[138] J. Vázquez-Salceda, H. Aldewereld, and F. Dignum. Implementing Norms in Multiagent Systems. In *Proceedings of Multiagent System Technologies, Second German Conference, MATES*, pages 313–327, 2004.

[139] D. Villatoro, G. Andrighetto, J. Sabater-Mir, and R. Conte. Dynamic sanctioning for robust and cost-efficient norm compliance. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Spain, 16-22 July 2011*, 2011.

[140] D. Villatoro, S. Sen, and J. Sabater-Mir. Topology and memory effect on convention emergence. In *Proceedings of the 2009 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technologies*, pages 233–240. IEEE, 2009.

[141] Daniel Villatoro, Giulia Andrighetto, Jordi Sabater-Mir, and Rosaria Conte. Dynamic sanctioning for robust and cost-efficient norm compliance. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence - Volume Volume One*, pages 414–419, Barcelona, 2011. AAAI Press.

[142] Daniel Villatoro, Jordi Sabater-Mir, and Sandip Sen. Social instruments for robust convention emergence. In Toby Walsh, editor, *Proceedings of the 22nd International Joint Conference on Artificial Intelligence - Volume Volume One*, pages 420–425, Barcelona, 2011. IJCAI/AAAI, AAAI Press.

[143] A. Walker and M. Wooldridge. Understanding the Emergence of Conventions in Multi-Agent Systems. In V. Lesser, editor, *Proceedings of the First International Conference on Multi-Agent Systems*, pages 384–389. MIT Press, 1995.

[144] Chi Wang, Hongbo Wang, Yu Lin, and Shanzhi Chen. A currency-based P2P incentive mechanism friendly with isp. In *ICCDA 2010: Proceedings of the International Conference on Computer Design and Applications*, volume 5, pages 403–407, 2010.

[145] C. J. C. H. Watkins and P. Dayan. Q-learning. *Machine Learning*, 8(3-4):279–292, 1992.

[146] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, 1998.

[147] A. Whitby, A. Jøsang, and J. Indulska. Filtering out unfair ratings in bayesian reputation systems. *The Icfain Journal of Management Research*, 4(2):48–64, 2005.

[148] M. Wooldridge. *An Introduction To MultiAgent Systems*. John Wiley & Sons Ltd., Chichester, 2002.

[149] M. Wooldridge and N. R Jennings. Agent theories, architectures, and languages: A survey. *Wooldridge and Jennings Eds., Intelligent Agents, Berlin: Springer-Verlag*, pages 1–22, 1995.

[150] F. López y López and M. Luck. Modelling norms for autonomous agents. In E. Chávez, J. Favela, M. Mejía, and A. Oliart, editors, *Proceedings of The Fourth Mexican Conference on Computer Science*, pages 238–245. IEEE Computer Society, 2003.

[151] F. López y López, M. Luck, and M. d'Inverno. A Normative Framework for Agent-Based Systems. *Computational and Mathematical Organization Theory*, 12:227–250, 2006.

[152] T. Yamashita, K. Izumi, and K. Kurumatani. An investigation into the use of group dynamics for solving social dilemmas. In P. Davidsson, B. Logan, and K. Takadama, editors, *Multi-Agent and Multi-Agent-Based Simulation*, volume 3415 of *Lecture Notes in Artificial Intelligence*, pages 185–194. Springer, 2005.

[153] M. Yang, Z. Zhang, X. Li, and Y. Dai. An empirical study of free-riding behavior in the maze P2P file-sharing system. In *IPTPS '05: Proceedings of the 4th International Workshop on Peer-to-Peer Systems*, pages 182–192. Springer, 2005.

[154] B. Yu and M. P. Singh. An evidential model of distributed reputation management. In *AAMAS '02: Proceedings of the 1st International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 294–301. ACM, 2002.

[155] G. Zacharia, A. Moukas, and P. Maes. Collaborative Reputation Mechanisms in Electronic Marketplaces. In *HICSS '99: Proceedings of the 32nd Annual Hawaii International Conference on System Sciences-Volume 8*, page 8026. IEEE Computer Society, 1999.